OPERATIONAL PERFORMANCE ENHANCEMENT OF HUMAN OPERATED FLEXIBLE SYSTEMS

A Thesis
Presented to
The Academic Faculty

by

Khalid Lief Sorensen

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
Woodruff School of Mechanical Engineering

Georgia Institute of Technology
August 2008

OPERATIONAL PERFORMANCE ENHANCEMENT OF HUMAN OPERATED FLEXIBLE SYSTEMS

Approved by:

Dr. William Singhose, Advisor
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Thomas Morley
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Stephen Dickerson
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Dr. Bruce Walker
School of Psychology
Georgia Institute of Technology

Dr. Ye-Hwa Chen
Woodruff School of Mechanical Engineering
Georgia Institute of Technology

Date Approved: 6 June 2008

To my parents, whose sacrificial support, unconditional love, and exemplary work ethic have, throughout the decades of my life, sustained and motivated. Thank you.

ACKNOWLEDGEMENTS

The collaborative efforts of faculty, students, and industrial partners are embodied in this dissertation. I extend, wholeheartedly, my gratitude to these numerous contributors.

My advisor, Dr. William Singhose, has provided insights, shaped by pragmatic experience, which have served to guide all aspects of this research. Additionally, his foundational work in the field of command shaping provided the framework upon which this work was constructed. I have also tremendously appreciated our frequent discussions on life, weightlifting, and vegetables.

Doctors Ye-Hwa Chen, Steven Dickerson, Thomas Morley, and Bruce Walker served as committee members, providing valuable perspectives on math, controls, and human factors. My collaboration over the years with Dr. Keith Hekman have also contributed to the completion of this work.

Help provided to me by my graduate and undergraduate collogues permitted the completion of far more work than would have been completed otherwise. These people include: Philipp Andersch, Patrick Cross, Aayush Daftari, Jon Danielson, Johannes Fisch, Johathan Fonseca, Terry Hall, John Huey, Dooroo Kim, Jason Lawrence, Stefan Locher, Joshua Spiers, Christoph Staheli, Jurg Suter, Brandon Terrell, Josh Vaughan, and Rolf Weiss. It has been a delight working with these individuals.

Several industrial partners were instrumental in the successful completion of this work. CAMotion provided financial support. Siemens Energy & Automation provided necessary software and hardware. Logan Aluminum provided access to industrial machines. I also extend my thanks to Gerald Greene, the facilities manager of the Georgia Tech Manufacturing Research Center, for his assistance in experimentation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Recent decades have been witness to explosive leaps in manufacturing productivity. Advances in communication technology, computing speed, control theory, and sensing technology have been significant contributors toward the increased productivity and efficiency that industry has exhibited. The continued growth of technological equipment and engineering knowledge challenges engineers to fully utilize these advancements in more sophisticated and useful automation systems.

One such application involves enhancing bridge and gantry crane operation. These systems are used throughout the globe, and are critical aspects of industrial productivity. Consequently, improving the operational effectiveness of cranes can be extremely valuable.

Effective control of cranes can be largely attributed to two distinct, but related aspects crane manipulation: 1) the expertise of operators, which are responsible for issuing commands to the structures, and 2) the dynamic properties of cranes, which influence how the structures respond to issued commands. Accordingly, the operational efficiency of cranes can be influenced by changing both the way that operators issue commands to cranes, and also how the crane responds to issued commands.

This thesis is concerned with dynamic control theory of flexible machines, and human/machine interaction, especially as these areas relate to industrial crane control. In the area of dynamic control, this thesis investigates control strategies that are specifically suited for use on systems that possess common actuator nonlinearities, like saturation, rate limiting, dead-zone, backlash, and finite-state actuation. In the area of human/machine interaction, this thesis investigates the effects of different crane interface devices on the operational efficiency of cranes.

CHAPTER I

CONTROL AND HUMAN FACTORS IN AUTOMATION

Industrial automation is a challenging endeavor. It requires the synergistic combination of process equipment with electronic control and human interaction. Effective automation has facilitated rapid advancements in the areas of transportation, manufacturing, and agriculture, while also proliferating increasingly affordable goods and services, improving industry working conditions, and increasing safety.

Two distinct, yet related, areas of industrial automation are addressed in this work: 1) dynamic control, and 2) human factors in machine operation. These aspects of industry are of interest because they contribute directly toward safety, efficiency, and usability in many industrial automation systems.

In the area of dynamic control, this thesis is concerned with a branch of control theory called *input shaping*. Input shaping is a filtering technique used to suppress motion-induced oscillation in flexible systems. Given that all physical systems exhibit some degree of compliance, input-shaping control has an inexhaustible application space. This research investigates the application of input shaping on systems that possess frequently encountered actuator nonlinearities, such as saturation, rate limiting, dead-zone, backlash, and other discontinuous behavior. Because all mechanical systems possess, in varying degrees, some of these nonlinear attributes, two fundamental research questions are motivated:

- *What are the effects of actuator dynamics on input-shaping control?*
- *What strategies can mitigate detrimental effects of actuator dynamics on input-shaping control?*

These research questions are addressed in the context of industrial bridge and gantry crane control.

In the area of human factors, this research is concerned with the human/machine interface of industrial cranes that utilize dynamic control in general, and input shaping in particular. The effects that different human/machine interface devices have on the operational efficiency of cranes equipped with dynamic control will be investigated. The fundamental research question driving this thrust of the investigation is:

- *Can the operational efficiency of cranes be improved beyond what is achievable through dynamic compensation alone by changing the way operators interact with a crane?*

The motivation for directing the research toward bridge and gantry cranes is that these systems occupy a crucial role within industry. They are used throughout the world in thousands of shipping yards, construction sites, steel mills, warehouses, nuclear power and waste storage facilities, and other industrial complexes. The timeliness and effectiveness of this manipulation system are important contributors to industrial productivity. Thus, research directed at investigating and improving the operational effectiveness of cranes can be extremely valuable. Additionally, these systems are ideal test beds for advanced control strategies because their flexible dynamics, and inherent nonlinearities, are well suited for evaluating the performance of different input-shaping techniques.

To facilitate an understanding of the presented research, and to more thoroughly motivate the need to address the posed fundamental research questions, a brief description of industrial cranes and the difficulties associated with their efficient operation is presented in Section 1.1. Section 1.2 introduces the theory of input shaping, and the problems that can arise when input shaping is applied to nonlinear systems. The information presented in this section is intended to provide the reader with a basic understanding of fundamental input-shaping principles. Section 1.3 presents a survey of literature dealing with areas relevant to input shaping and crane control. Existing knowledge gaps in the current state-of-the-art are also identified. Finally, Section 1.4 describes how the remainder of this thesis is organized. It also delineates the primary contributions of this work.

1.1 Challenges of Crane Manipulation

Operational efficiency, in a general sense, is a measure of productivity under normal operating circumstances. It is a broad metric that seeks to capture the various nuances of operation. There are several characteristics of industrial cranes and their use that contribute to their operational efficiency. These attributes may be broadly grouped into three categories: dynamic response, task difficulty, and the human factor.

1.1.1 Dynamic Response

One inherent property of cranes is particularly detrimental to efficient operation: the tendency for a payload to swing. As a result, tremendous effort has been directed at improving the operational efficiency of cranes by developing and implementing dynamic compensation schemes. These techniques have included various combinations of feedback control, feedforward control, open-loop filtering, optimal trajectory generation, and other oscillation suppression methods. The common objective of these control methods is to reduce a crane's oscillatory response to both issued commands and/or external disturbances.

1.1.2 Task Difficulty

The inherent difficulty of a given crane manipulation task also contributes to operational efficiency. For example, at the Hanford Site in Washington State, crane operators routinely stack radiological packages in tight matrix formations, requiring placement precision on the order of a few centimeters. Because of the hazardous content of the payloads, and the required precision, operators are extremely careful to avoid obstacles. As a result, cranes are driven slowly to aid in achieving the desired placement and obstacle avoidance. On the other hand, crane operators in some steel mills, requiring only gross motion of scrap steel, are not overly concerned about obstacle avoidance or final positioning accuracy. Crane motions are made quickly in order to be efficient.

The variable nature of crane motion exemplified in these two examples illustrates how task difficulty affects the speed with with cranes are manipulated, and therefore also is an integral contributor to operational efficiency. In some cases, task difficulty is closely

associated with the inherent properties of the manipulation task, such as the desired path, path restrictions, and the dynamic response of the crane. In other cases, task difficulty depends on external factors, such as wind disturbances on the payload.

1.1.3 Human Factors

Another aspect of industrial crane manipulation that contributes to operational efficiency is related to the role of the human operator. To achieve a desired manipulation objective, a crane operator must be cognizant of several factors that influence his or her control decisions. These factors include the dynamic response of the motors and drives to issued commands, the response of the payload to motion of the structure, obstacle avoidance, and positioning error. Control decisions based upon constant monitoring of these factors are realized by an operator when he or she physically actuates the crane. Conventionally, this is accomplished through the use of a control pendent or actuating levers. Operational efficiency is affected by the proficiency of an operator at processing monitored information, making control decisions based upon this information, and actuation of the crane through an interface device.

1.1.4 Fundamental Research Question

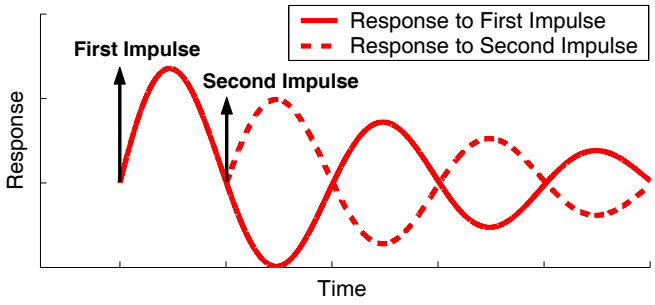
Much of the work directed at improving the operational efficiency of cranes has done so by implementing various dynamic compensation schemes to suppress the oscillatory response of the payload. This compensation has tended to reduce task difficulty because the human operator is less absorbed in monitoring and compensating for payload swing. As a result, he or she can direct more effort toward other necessary tasks. The fundamental research question pertaining to crane manipulation arises from this state-of-art. It is restated here:

- *Can the operational efficiency of cranes be improved beyond what is achievable through dynamic compensation alone by changing the way operators interact with a crane?*

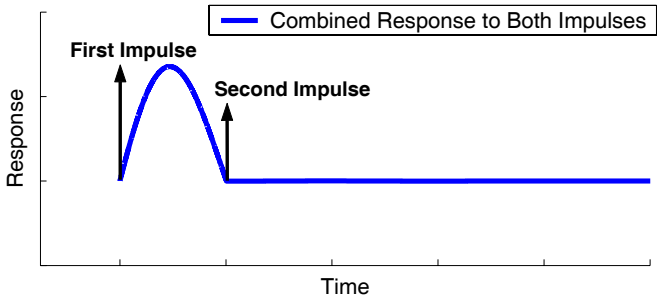
1.2 Capability and Challenges of Input Shaping for Oscillation Suppression

Input shaping is a command filtering technique [77]. It was designed for the purpose of reducing motion-induced oscillation in flexible systems. Reference commands modified by an input-shaping filter are issued to a system instead of unfiltered commands. The modified commands cause the system to cancel out its own motion-induced oscillation.

To demonstrate this fundamental interference principle, consider the response of a lightly damped second-order system to a series of two impulses, as shown in Figure 1.1. Figure 1.1(a) shows the system response to the impulses if they were each applied individually. Figure 1.1(b) shows the combined system response resulting from superposition. These figures demonstrate that impulses of appropriate amplitudes and times can be applied to a system in such a way that the resonant modes of the system combine destructively, resulting in zero residual oscillation.



(a) System Response to Each Impulse.



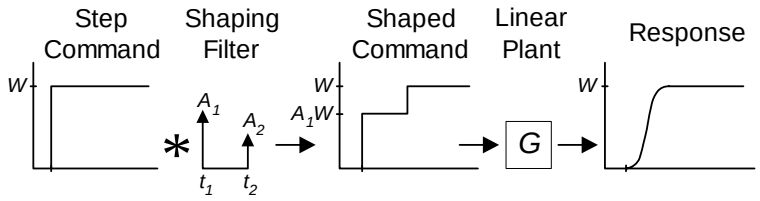
(b) System Response to Both Impulses.

Figure 1.1: Destructive Interference Caused by a Sequence of Impulses.

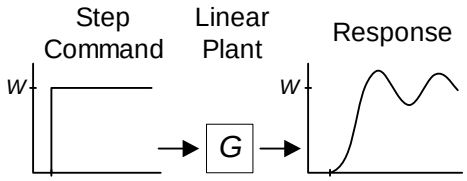
The input-shaping process is an extension of the concept illustrated in Figure 1.1 that

allows arbitrary reference commands, rather than just sequences of impulses, to produce low residual oscillation. This extension is made by convolving an appropriate impulse sequence, known as an input shaper, with a given reference command to produce a shaped command. The shaped command has the same oscillation-reducing properties as the original set of impulses.

An example of the input-shaping process for a two-impulse input shaper is illustrated in Figure 1.2(a). After the reference command (in this case a step) is convolved with the impulse sequence, a shaped command is produced. When actuated by the shaped command, the linear plant, G, exhibits zero oscillation. For comparison purposes, Figure 1.2(b) illustrates the response of the plant when input shaping is not used.



(a) Shaped Command Actuating a Linear Plant.



(b) Unshaped Command Actuating a Linear Plant.

Figure 1.2: Shaped and Unshaped Command Actuating a Linear Plant.

A more general representation of the input-shaping process is shown with the block diagram of Figure 1.3. Here, a reference command, $x(t)$, is modified by an input-shaping filter, IS, to produce the shaped command, $y_s(t)$. The linear system, G, responds to the shaped command with $y_G(t)$. The convolution operation used to produce the shaped command can be represented in the time domain as:

$$y_s(t) = IS(t) * x(t) \tag{1.1}$$

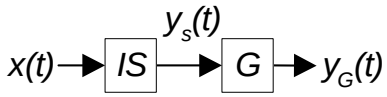


Figure 1.3: General Input-Shaping Process.

A general n-impulse input-shaping filter can be expressed in the time domain as:

$$IS(t) = A_1\delta(t) + \sum_{i=2}^{n} A_i\delta(t - t_i) \tag{1.2}$$

where $0 < t_i < t_{i+1}$, and $A_i \neq 0$. The symbol $\delta(t)$ is the Dirac delta function, A_i is the amplitude of the i^{th} impulse, and t_i is the time of the i^{th} impulse. If $x(t) = 0$ for $t < 0$, then the expression for $y_s(t)$ may be expanded by substituting (1.2) into (1.1) yielding:

$$y_s(t) = A_1x(t) + A_2x(t - t_2) + \cdots + A_nx(t - t_n) \tag{1.3}$$

In general, for finite valued reference commands that reach a steady-state value, and for input shapers with correctly selected impulse amplitudes and times, a linear time-invariant system can always exhibit arbitrarily low residual oscillation in response to $y_s(t)$.

The impulse amplitudes, and the time durations between successive impulses in an input shaper influence the degree of vibration suppression, the frequency ranges over which the filter is effective, and total duration of the filter. There are a number of different input-shaping filters that possess these attributes in varying degrees. Frequently used input shapers include the two-impulse zero-vibration (ZV) shaper [77,96], the three-impulse zero-vibration and derivative (ZVD) shaper [77], and the three-impulse unity-magnitude and zero-vibration (UMZV) shaper [80]. The advancements made in this thesis to the field of input-shaping theory are discussed in the context of ZV, ZVD, and UMZV input shapers. Therefore, the derivation of these common input-shaping filters is discussed in the following section.

1.2.1 Input Shaper Constraint Equations

The amplitudes and time locations of the impulses in an input shaper can be determined by solving a set of constraint equations [24]. The constraints can be categorized as 1) constraints on the impulse amplitudes, 2) residual vibration constraints, 3) requirement of time optimality, and 4) robustness constraints.

A constraint on the impulse amplitudes is necessary to ensure that the shaped command achieves the same steady-state value as the original reference command. This constraint is that the sum of the impulse amplitudes must equal one:

$$\sum_{i=1}^{n} A_i = 1 \tag{1.4}$$

The constraints on residual vibration limit the oscillation induced into a system by the shaped command. These constraints can be expressed as the ratio of two physical quantities: the residual oscillation amplitude induced into a system by a shaped command divided by the residual oscillation amplitude caused by the unshaped counterpart. This ratio is given by [77]:

$$V(\omega, \zeta) = e^{-\zeta\omega t_n}\sqrt{C^2 + S^2} \tag{1.5}$$

where,

$$C(\omega, \zeta) = \sum_{i=1}^{n} A_i e^{\zeta\omega t_i} \cos(\omega\sqrt{1-\zeta^2}t_i) \tag{1.6}$$

$$S(\omega, \zeta) = \sum_{i=1}^{n} A_i e^{\zeta\omega t_i} \sin(\omega\sqrt{1-\zeta^2}t_i) \tag{1.7}$$

The symbols ω and ζ are the frequency and damping ratio, respectively, over which the vibration constraint holds. Ordinarily, these parameters are set equal to the natural frequency and damping ratio of the system for which the input shaper is being used. However, because *exact* values of these system parameters cannot be known, modeled estimates are used instead. The input shaping parameters, A_i and t_i, are obtained by solving (1.4) and (1.5) when V is constrained to be less than or equal to a tolerable level of vibration, V_{tol}.

Robustness constraints are used to ensure that an input shaper will be robust to uncertainty in the modeled parameters. Many different kinds of robustness constraints exist. One such constraint requires the partial derivative of (1.5) with respect to frequency to be equal to zero at the modeled parameters [77]. Using this constraint, significant vibration suppression occurs even when there are large modeling errors. This constraint may be represented as:

$$\frac{\partial}{\partial\omega}\left[e^{-\zeta\omega t_n}\sqrt{C^2 + S^2}\right] = 0 \tag{1.8}$$

Due to the transcendental nature of (1.5), there are multiple input shapers satisfying the vibration and robustness constraints. The time optimality constraint obtains the shortest duration shaper by minimizing t_n, the time of the last impulse. Minimizing t_n is desirable because it results in a fast rise time for the shaped command.

1.2.2 Positive and Negative Input Shapers

Additional constraints on the impulse amplitudes are necessary in order to place a limit on the magnitude of the impulses. This is because minimizing the time duration of the input shaper drives the impulse amplitudes to positive and negative infinity.

One such constraint requires that the input shaper be comprised only of positive impulses. Accordingly, this class of filters is called *positive input shapers*. An alternative constraint permits the impulses to assume negative, but finite, values. Therefore, this class of filters is called *negative input shapers* [80].

1.2.3 ZV-Shaper

A two-impulse shaper is obtained by solving (1.4) and (1.5) when V is set equal to zero. Accordingly, this filter is aptly named the zero-vibration, or ZV, input shaper [96, 77]. The impulse amplitudes and times are:

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{1+K} & \frac{K}{1+K} \\ 0 & \pi/\omega_d \end{bmatrix} \tag{1.9}$$

where,

$$K = exp\left(\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}\right)$$

and,

$$\omega_d = \omega_n\sqrt{1-\zeta^2}$$

1.2.4 ZVD-Shaper

While the ZV shaper performs well when the modeled frequency is at or very near the actual system frequency, large modeling errors can result in unacceptably high oscillation. An input shaper that is more robust to frequency uncertainty can be obtained by satisfying the derivative constraint of (1.8), in addition to (1.4) and (1.5) when V is set equal to zero.

Accordingly, this filter is named the zero-vibration and derivative, or ZVD, shaper [77]. The impulse amplitudes and times are:

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{(1+K)^2} & \frac{2K}{(1+K)^2} & \frac{K^2}{(1+K)^2} \\ 0 & \pi/\omega_d & 2\pi/\omega_d \end{bmatrix} \tag{1.10}$$

1.2.5 UMZV-Shaper

The ZV and ZVD input shapers consist of positive impulses. Shorter duration input shapers may be obtained if the impulse amplitudes are permitted to assume negative values [80]. If the impulse amplitudes are constrained to equal 1 or -1, then the shaper can be used to produce time-optimal or fuel-optimal commands [50, 79, 83]. A frequently encountered negative input shaper is obtained by enforcing the unity-magnitude constraint, and solving (1.4) and (1.5) when V is set equal to zero. The unity-magnitude zero-vibration, or UMZV, shaper may be expressed as [80]:

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 0 & t_2 & t_3 \end{bmatrix} \tag{1.11}$$

where,

$$\begin{bmatrix} t_2 \\ t_3 \end{bmatrix} \cong \begin{bmatrix} (2\pi/\omega_d)(1/6 + 0.272\zeta + 0.203\zeta^2) \\ (2\pi/\omega_d)(1/3 + 0.005\zeta + 0.179\zeta^2) \end{bmatrix}$$

The expression in (1.11) is the product of curve fitting numerical solutions for the impulse times. It provides the exact solutions for t_2 and t_3 when $z = 0$. However, the resulting impulse times for lightly damped systems are within 0.5% of their actual values over the range of $0 < \zeta \leq 0.3$. Alternative equations for the impulse times are derived in [25], which are valid for higher damping ratios.

1.2.6 Graphical Shaper Representation

The impulse sequence comprising an input shaper can be graphically represented in the time domain as a train of impulse functions. Graphical representation of the ZV, ZVD, and UMZV input shapers are depicted in Figure 1.4.

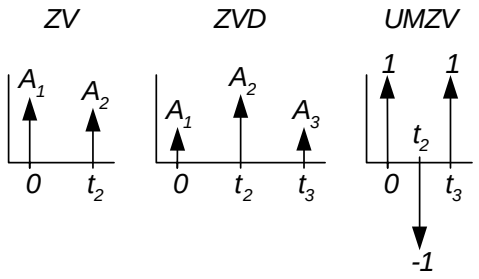


Figure 1.4: Graphical Representation of the ZV, ZVD, and UMZV Input Shapers.

1.2.7 Input Shaping on Linear and Nonlinear Systems

The effectiveness of input shaping at suppressing unwanted oscillation in cranes was demonstrated in [41]. Experimental results from this study are shown in Figure 1.5. In this experiment, an operator manipulated the payload of a 10-ton industrial bridge crane with and without a ZV input-shaping filter. The objective was to move the payload from a starting region to a target region, while attempting to avoid stationary barriers. The results shown in Figure 1.5 depict the motion of the payload in response to the operator commands, as measured by an overhead camera.

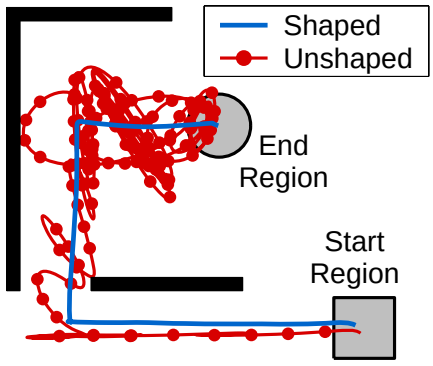


Figure 1.5: Shaped and Unshaped Payload Motion From a 10-Ton Bridge Crane.

When input shaping was not enabled, the operator commands were issued directly to the crane. Payload swing was induced into the system when the crane was commanded to move, change the direction of travel, and commanded to stop. These motions are shown with the dotted line. When input shaping was enabled, the operator commands were modified by the shaping filter before being issued to the crane. The payload trajectory resulting from the filtered commands is shown with the solid line. This trajectory is generally free from

oscillation.

The oscillation suppression demonstrated by these results is typical for systems that behave in a linear, or nearly linear manner. However, the capacity of a system to behave in a linear manner depends largely upon how the system is actuated. Many types of motors or drive systems exhibit nonlinear behavior. Typically, actuator nonlinearities can be characterized by discontinuous nonlinear elements such as saturation, rate limiting, dead-zone, and backlash. As stated previously, virtually all mechanical systems possess, in varying degrees, some of these nonlinear attributes. For example, systems cannot be driven infinitely fast, nor be accelerated instantaneously. These attributes constitute saturation and rate limiting in the velocity domain, respectively. Consequently, the block diagram shown previously in Figure 1.3 is an over simplified representation of how input shaping is implemented on a physical system because it does not account for the behavior of the actuation process. In constructing this block diagram, it was assumed that the controlled plant precisely follows the shaped commands.

A more accurate model of how input shaping is implemented on a physical system is represented in the block diagram of Figure 1.6. The behavior of the actuating *drives* and *motors* is captured in the block, DM. Note that rather than the shaped signal, $y_s(t)$, actuating the plant, the signal, $y_{np}(t)$, actuates the plant. If DM exhibits significant nonlinear behavior, then $y_{np}(t)$ can be corrupted to the extent that the vibration reducing properties of $y_s(t)$ are substantially reduced.

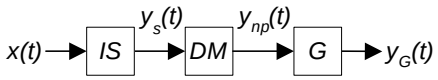


Figure 1.6: General Input-Shaping Process with Drives and Motors Block.

1.2.8 Fundamental Research Question

Generally speaking, the conventional procedure for formulating and implementing input shaping control neglects or oversimplifies the actuation behavior of the controlled system. Consequently, implementation of input shaping on systems that exhibit significant actuator nonlinearities has resulted in unanticipated and undesirable system performance. This

circumstance, coupled with the ubiquitous presence of nonlinear systems in industry, motivates the fundamental research questions pertaining to input-shaping control. They are restated here:

- *What are the effects of actuator dynamics on input-shaping control?*
- *What strategies can mitigate detrimental effects of actuator dynamics on input-shaping control?*

1.3 Survey of Prior Research

There are several areas within the fields of control and human factors that are relevant to the research presented in this thesis. The objective of this section is to discuss the current state-of-the-art in these areas and identify existing knowledge gaps.

1.3.1 Input Shaping

The conception of input shaping theory is most often attributed to the work O.J.M. Smith in the late 1950's [96]. Originally named "Posicast Control", this shaping technique involved resolving a baseline command into two identical, but scaled components. These were subsequently offset in time and then summed to produce a shaped command. The modification technique was effective at reducing oscillation when the linear system model matched closely with the actual system [117]. However, the implementation of this control method was prohibitively difficult to implement on the analogue controllers of the time.

1.3.1.1 Input Shaping For Linear Systems

There have been many developments in the area of input shaping since Posicast Control. However, the first significant advancements occurred roughly thirty years after it was first introduced to the controls community. The proliferation of digital control systems, which enabled the technique to be practically implemented, helped spur these advancements.

The first significant advancement in terms of robustness to modeling errors was the development of the ZVD input shaper by Singer and Seering [77]. Immediately following their development, a graphical vector diagram approach to representing input shapers was developed [92]. The insight gained through this graphical representation of input shaping led to

other significant input-shaping breakthroughs. These include input shapers for multi-mode systems [33, 63, 90], and input shapers with a specified amount of robustness to modeling errors [93, 60, 61].

One beneficial attribute of input shaping is the relative ease with which it can be implemented on computer controlled, or PLC-based systems. For this reason, many industrial systems have incorporated input shaping into their control architecture. For example, the performance of cranes [76, 99, 9, 108, 105], coordinate measuring machines [74, 88, 35], satellites [120, 91], micro-milling machines [20], disk-drive manufacturing machines [80], silicone handling robots [69], magnetic tape data readers [13], and long-reach manipulators [51, 54, 24], have been significantly improved with input shaping.

1.3.1.2 Input Shaping For Nonlinear Systems

Input shaping was developed using linear system theory. Therefore, the degree to which it may be successfully applied to a nonlinear system depends, in large part, on the degree to which the nonlinear system behaves as a linear one. In spite of this apparent limitation on input shaping, this technique has been used on many systems with nonlinear dynamics. The different strategies employed to accomplish this may be broadly grouped into three categories: 1) robustness approach, 2) adaptive approach, and 3) nonlinear formulation.

Robustness Approach: The robustness approach has been utilized on nonlinear systems where the principal effect of the nonlinearity is to cause slight changes in the natural frequency of the system. The robustness approach makes use of input shapers that are non-sensitive to these changes. The ZVD shaper was the first robust input shaper [77]. Later, the *extra insensitive*, or EI shaper was developed [87]; this shaper exceeds the robustness of the ZVD shaper without increasing its duration. *Specified insensitivity*, or SI, shapers allow an arbitrary level of robustness to frequency shifts [93].

The robustness approach to nonlinear input shaping has worked well on structures that exhibit moderate shifts in system frequency. However, for systems with widely varying frequencies, the robustness approach yields input shapers that cause slow system responses. Therefore, adaptive approaches have been developed.

Adaptive Approach: Like the robustness approach, the adaptive approach is suited for nonlinear systems in which the nonlinearity exhibits itself primarily by changing the system frequency. Additionally, this approach is also suitable when the nonlinearity causes changes in the damping of a system. The technique makes use of sensory information to determine the current state of a given system. Using this information, input shaper parameters are adapted to reduce oscillation. Several researchers have employed this technique [10, 42, 52, 65, 71, 123, 115].

Nonlinear Formulation: The robustness and adaptive approaches to input shaping on nonlinear systems are similar to one another in that they both rely on a linearized version of the nonlinear system for which they are being used. In contrast, nonlinear formulations are based upon a nonlinear system model. Research using nonlinear formulations include Kinceler, et. al [45], Gorinevsky, et. al [23], and Smith et. al [95]. The nonlinear dynamics of the systems addressed by these authors is attributed to continuous (differentiable) nonlinearities, such as those encountered in the complex kinematics of multi-link robotic arms.

Nonlinear dynamics can also be attributed to discontinuous nonlinearities within a system, such as backlash, saturation, rate limiting, and dead-zone. When a hard nonlinearity modifies an input-shaped command, the vibration reducing properties of the shaped command can be significantly degraded. Lawrence et. al considered input shaping from the perspective of backlash [47] and coulomb friction [48]. Though recent, Lawrence's work represents some of the earliest considerations of hard nonlinearities in an input-shaping control system. He proposed techniques of input shaper redesign to reduce the detrimental effects of the nonlinearities. Danielson et. al used the analysis and mitigation techniques of Lawrence to analyze rate limiting for a limited number of input shapers [14].

1.3.1.3 Knowledge Gap

Relatively little research has been conducted that addresses nonlinear input shaping. Even less has been directed at examining the special cases of hard nonlinearities. A unified framework for evaluating the effects of nonlinearities on input-shaped commands has not

been developed. Additionally, the proposed mitigation strategies for reducing the effects of hard nonlinearities on input shaping have not been systematically developed, nor are they generally applicable.

1.3.2 Hard Nonlinearities

Hard nonlinearities such as saturation, rate limiting, dead-zone, and backlash are pervasive in mechanically actuated systems. These types of common hard nonlinear elements have been studied extensively and are well understood. The functional relationship between the input and the output of saturation [73, 58, 121], rate limiting [4], dead-zone [119], backlash [118, 34, 47], and Coulomb friction [48] have been described in detail.

1.3.2.1 Knowledge Gap

Some hard nonlinear elements have not been modeled or mathematically characterized, yet their presence is evident in the response of some systems.

1.3.3 Closed-Loop Signal Shaping

Traditionally, input shaping has been utilized in an open-loop manner, like the block diagram of Figure 1.3. Given that input shaping is a successful method for reducing unwanted oscillations in this open-loop configuration, it is natural to investigate the potential benefits of utilizing input shaping within a feedback loop.

The research that addresses in-the-loop input shaping can be broadly categorized into three groups [31]: 1) feedforward signal shaping, 2) model reference signal shaping, and 3) plant inversion signal shaping. The distinction between these three groups is made based upon the general block diagram form assumed by each of these three groups. The category of closed-loop input shaping that is of primary concern to the research presented here is that of feedforward signal shaping. The general block diagram form for this type of in-the-loop shaping is shown in Figure 1.7.

Notice that the input shaper filters signals generated by the control block, and then issues the filtered signals to the controlled plant. This type of closed-loop input shaping is categorized in the feedforward group because the input shaper is in the feedforward path

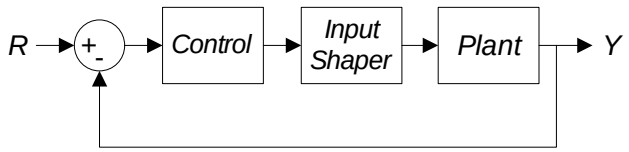


Figure 1.7: Feedforward Closed-Loop Input Shaping.

of the block diagram.

Several researchers have investigated different aspects of this type of closed-loop signal shaping (CLSS). Kapila, et. al [36, 37] and Tzes [122] considered the robustness of CLSS controllers to modeling errors. Zuo, et. al, and Drapeau, et. al compared the performance of CLSS controllers to that of proportional-integral-derivative (PID) controllers and standard open-loop input shaping controllers [15, 125, 126].

Because input shaping in the loop causes a partial time delay, stability of the CLSS design is a primary concern. This issue has also been addressed [54, 36, 37, 125, 126, 11, 53, 116]. Perhaps the most comprehensive work on CLSS was conducted by Huey [31]. This work advanced the fundamental understanding of CLSS controllers in terms of stability, robustness, utility, and performance, while also validating other researchers results through experimental and simulation testing.

1.3.3.1 Knowledge Gap

A fundamental understanding of CLSS controllers as applied to systems with hard nonlinear elements is lacking.

1.3.4 Dynamic Crane Control

A large amount of research pertaining to the control of cranes has been conducted. Engineers have sought to improve the ease-of-use, increase operational efficiency, and mitigate safety concerns by addressing three primary aspects of crane systems: 1) motion-induced oscillation, 2) disturbance-induced oscillation, and 3) positioning capability. The control schemes developed to address these areas may be broadly grouped into three categories: 1) command synthesis, 2) command filtering, and 3) feedback control. The following review of crane control literature is not comprehensive, but is representative of the work conducted

in these categories.

Command Synthesis: Command synthesis is the process of precomputing partial or complete segments of actuator commands based on *a-priori* or learned knowledge about a given system. Optimal control typically belongs to this category. Piazzi proposed a dynamic-inversion-based and optimal control for reducing transient and residual motion-induced oscillation [68]. O'Connor developed a control strategy based on mechanical wave concepts that involves learning unknown dynamics through an initial trolley motion. Once the control estimates the unknown dynamics, commands are synthesized that suppress induced oscillation [59]. One drawback related to command synthesis techniques is their inability to be implemented in real time, owing to the necessity of precomputation of system trajectories. There is no known implementation of this type of control scheme used with a commercial crane [26].

Command Filtering: Command filtering is a reference signal modification technique that is implementable in real time. However, command filtering does not have the closed-loop mechanisms of feedback control, and must, therefore, be used in conjunction with a feedback control if it is to be used for disturbance rejection.

Some of the earliest work attempting to reduce motion-induced oscillation was performed by Alsop et. al using phase-plane superposition principles [2]. Starr implemented swing-free transport of suspended payloads on a PUMA robot by generating staircase velocity profiles similar to those obtained by shaping step commands with a ZV input shaper [114]. Singer et. al reduced motion-induced oscillations of a 15-ton bridge crane by using robust input-shaping techniques [76]. Singhose et. al implemented multi-mode input shaping filters on a 10-ton bridge crane [85].

Feedback Control: Feedback control is a common strategy to mitigate positioning and cable sway errors. This type of control is aptly suited for positioning a bridge or trolley. However, when a feedback controller must minimize cable sway, the control task becomes much more problematic. Accurate sensing of the payload must be implemented, which is often costly or difficult. These difficulties are discussed in detail for fully automatic commercial cranes in use at the Pasir Panjang terminal in Singapore [26]. Furthermore,

feedback control schemes are somewhat slow because they are inherently reactive. For example, when feedback is utilized to control cable sway, cable sway must be present in the system before the control will attempt to eliminate the undesired oscillations.

Moustafa developed nonlinear control laws for payload trajectory tracking based on a Lyapunov stability analysis [56]. Fang et. al proposed to control final trolley position and motion-induced oscillation through proportional-derivative (PD) control in which the coupling between the cable angle and the motion of the trolley was artificially increased [17]. Kim implemented a pole-placement strategy on a real container crane to control motion and disturbance-induced oscillation, as well as final positioning [44]. Finally, Fliess used a generalized state variable model suggested in [18], and then proposed a linearizing feedback control law [19]. The position of the trolley and length of the payload cable were the controlled variables, and their respective reference trajectories were limited to a class of fourth-order polynomials to insure minimal payload sway. The control did not attempt to eliminate disturbance-induced oscillations.

1.3.4.1 Knowledge Gap

Little research effort has been directed at developing a dynamic crane control that utilizes the strengths of command shaping together with the strengths of feedback control.

1.3.5 Human-Machine Interaction

The interaction between the human operator and an automation system is important in terms of safety and operational efficiency. A tremendous amount of work pertaining to the optimal design of human/machine interfaces has been completed, and is, in fact, the subject of an entire field of research. For the scope of the research presented here, a comprehensive literature review of human/machine interface devices in not possible. What is presented are some significant findings in this field that pertain, or may be readily applied to, crane control.

Sheridan developed a supervisory control model for automation systems that describes the modes of interaction between a human operator and the controlled system [75]. A complementary interaction model was developed by Stahre [113] that divided the process

control system into two interconnected processes. As applied to cranes, these are the human interface and the crane instruction processes. Rasmusses associated operator tasks with operator cognitive behavior [70]. An informative and descriptive framework for evaluating operator tasks and associated cognitive behaviors is the cognitive activity demand matrix, developed by Hollnagel [28]. Amat [3], and Frigola [21], emphasized the importance of gesture-based control because of its intuition-based cognitive demand. Finally, Fujita distinguished between skilled and unskilled human operators based upon in-the-loop operational ability [22].

1.3.5.1 Knowledge Gap

Established theory pertaining to general human/machine interaction has not been rigourously applied to the special case of human crane operation. Therefore, the advancement of interface devices for human crane control has not benefited from this inquiry. Additionally, methods to classify different types of crane motion and evaluate the effects of different interface devices on the operational performance of cranes is lacking.

1.4 Thesis Organization and Contributions

This thesis addresses the posed fundamental research questions and fills existing knowledge gaps by presenting the development and implementation of nonlinear input-shaping techniques, and developments from investigations of human factors in crane control. The body of this thesis presents this material in three parts: Chapter 2 through Chapter 6 present advanced input-shaping theory and implementation techniques used for systems with frequently encountered actuator nonlinearities; Chapter 7 and Chapter 8 discuss developments in the area of human/crane interaction; Chapter 9 and Chapter 10 present case studies of crane control that utilize the developed input-shaping techniques and human factors research; Chapter 11 contains concluding remarks. More specific descriptions of the thesis content follow:

1.4.1 Organization

Part 1: Nonlinear Input Shaping

Chapter 2 presents the development of a deconvolution analysis technique. This analysis method can be used to assess, both qualitatively and quantitatively, the oscillatory effects of arbitrary commands on linear second-order systems.

Chapter 3 utilizes the developments of Chapter 2 to characterize the detrimental effects of rate limiting, saturation, dead-zone, and backlash on systems that use ZV, ZVD, or UMZV input shapers.

Chapter 4 presents strategies to mitigate the detrimental effects of rate limiting, saturation, dead-zone, and backlash on systems that use ZV, ZVD, and UMZV input shapers.

Chapter 5 discusses the difficulties associated with input shaping on systems with a finite number of actuation states. Finite-state actuation describes systems that can only be propelled at discrete levels of actuator effort. This occurs on systems such as cranes that use relay-driven motors, and satellites with on-off thruster jets.

Chapter 6 examines a special case of saturation called overcurrenting. The source of overcurrenting in systems using input shaping is identified. The detrimental effects of overcurrenting, and a strategy to prevent this phenomenon, is described in the context of a theoretical construct called the *command space*. This construct provides insight into why overcurrenting occurs, and the effects that different mitigation strategies have on operational efficiency.

Part 2: Human Factors in Crane Control

Chapter 7 uses established human factors theory to develop models of human/crane interaction. These models serve to motivate the development of a new kind of crane interface that uses vision and touchscreen technologies. A thorough description of the interface is presented.

Chapter 8 examines the operational effects of different crane interfaces, including the visual touchscreen interface presented in Chapter 7. To accomplish this, a system for

classifying different types of crane motion is presented. The classification system provides a framework to assess the influence that the different interfaces have on efficiency and safety. The results from several operator experiments are presented in which the visual interface and conventional interface devices were tested.

Part 3: Case Studies of Industrial Crane Control

Chapter 9 presents a case study of crane control. A bridge crane at an industrial aluminum manufacturing plant, which exhibits predominantly single-mode payload dynamics, was equipped with a crane manipulation system. The system utilizes the developed nonlinear input-shaping control and human/machine interface in order to enhance safety, ease-of-use, and efficiency.

Chapter 10 extends the work presented in Chapter 9 by presenting another case study of industrial crane control. The crane considered in Chapter 10 exhibits more complicated payload dynamics than that of Chapter 9. A thorough dynamic analysis is presented and utilized in the design of the implemented crane manipulation system.

1.4.2 Contributions

This thesis advances the field of controls, in general, and the area of nonlinear input shaping, in particular. Additionally, the examination of crane manipulation in the context of human factors provides a fundamental understanding of the influence that human/machine interfaces have on operational efficiency. Specific contributions of this thesis are:

Contributions in the Area of Dynamic Control

- A generalized analysis technique is developed for assessing the oscillatory effects of arbitrary commands on linear second-order systems. This analysis technique is useful for quantifying the the oscillatory effects of actuator nonlinearities on systems using input shaping filters.
- A thorough understanding of the effects of several frequently encountered actuator nonlinearities on systems using common input shapers is obtained through theoretical and experimental results.

- Practical mitigation techniques are developed for eliminating or reducing the detrimental effects of frequently encountered hard nonlinearities on shaped signals.
- A more thorough understanding of CLSS and its utility for industrial systems is gained.

Contributions in the Area Human/Machine Interface Devices

- A visual human/machine interface device for commanding crane motion is developed and implemented on several industrial cranes.
- A classification and performance measurement system for categorizing different crane tasks and operator behavior is developed. This construct is used to gain understanding of how different interface devices affect the operational performance of cranes.

CHAPTER II

DECONVOLUTION ANALYSIS[1]

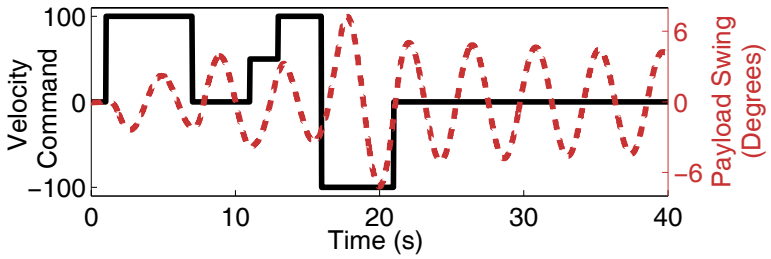
CHAPTER SUMMARY: *Input shaping is a well-established technique used for reducing the vibratory response of dynamic systems. Analytical tools are available for systems utilizing input shaping. These tools aid in performance analysis by providing intuitive and computationally simple methods for determining key system attributes, such as residual vibration in response to a command. This chapter describes methods whereby arbitrary reference commands may be interpreted as input-shaped commands. This capability allows input shaping analysis tools to be used on systems without input shapers. Experimental results obtained from an industrial bridge crane validate the theoretical developments.*

Reference commands used to drive flexible systems have a tremendous influence on performance. For example, consider the velocity command shown with the solid line in Figure 2.1(a). This command was generated by a human operator while driving a 10-ton industrial bridge crane. The dashed line in Figure 2.1(a) shows that the command induced a large amount of transient and residual swing in the crane payload.

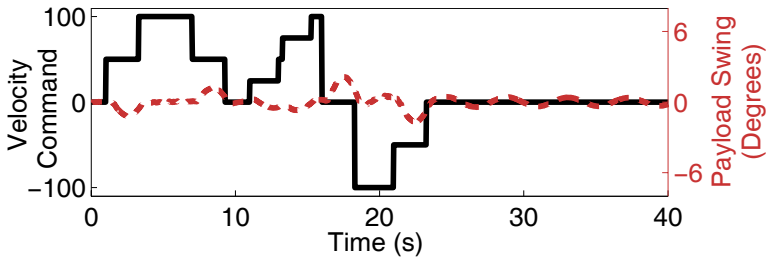
Now consider the velocity command in Figure 2.1(b). This command is similar in form to the command of Figure 2.1(a), but has more complicated amplitude variations. However, in spite of its more complicated appearance, the command induces much less oscillation, as shown by the dashed line in Figure 2.1(b). Because reference commands can produce such widely varying, and seemingly counterintuitive, responses, it is important to understand the qualities of a command that influence induced vibration.

This chapter investigates the vibration-inducing qualities of reference commands. The result of this investigation is an analytical analysis tool that is well suited for describing

[1]The contents of this chapter are published in [107].



(a) Command Inducing Noticeable Residual Oscillation.



(b) Command Inducing Minimal Residual Oscillation.

Figure 2.1: Payload Swing Response (Dashed) to Different Velocity Commands (Solid).

the effects that nonlinear elements have on systems using input shaping. This advancement will be used in subsequent chapters to quantify the oscillatory effects that several actuator nonlinearities have on frequently encountered input-shaping filters.

To develop the analysis technique, arbitrary reference commands are examined from the perspective of input-shaped commands. This is a desirable approach because the influence that shaped commands have on residual oscillation depends fundamentally on the input shaper, and not on the command itself. A more definitive statement of this relationship is:

> *The response of a linear plant to a shaped command is proportional to the response that would be induced by actuating the plant with the impulse sequence comprising the input shaper* [87, 16].

Therefore, a system's oscillatory response to a shaped command can be determined by considering only the input shaper, not the shaped command that actually drives the system.

Singhose et. al. has developed a vector diagram approach for analyzing residual system vibration induced by input shapers [87]. This analytical tool provides a graphical, intuitive, and computationally simple method for predicting vibratory system behavior in response to

input-shaped commands. The limitation that this tool can only be used with input-shaped systems serves as a motivation for the two primary objectives of this chapter:

1. *Interpret arbitrary commands as input-shaped commands.*

   By so doing, one may:

2. *Determine the oscillatory effects of arbitrary commands on linear systems by using vector diagram principles.*

Eloundou et. al. completed an initial investigation addressing these objectives [16]. Common "smooth" transitionary profiles were interpreted as shaped commands. However, this study was limited to S-curves, versines, and transition sine functions. This chapter expands the previous work to include *any function*, and thereby demonstrates that arbitrary commands and input-shaped commands can be analyzed in a universal manner.

To achieve the primary objectives set forth in this chapter, it will be shown that arbitrary signals can be deconvolved into both 1) a specified input shaper and reference command, and 2) a specified reference command and an input shaper. The former deconvolution is used to *qualitatively* describe the oscillation inducing attributes of arbitrary signals. The latter deconvolution, in conjunction with vector diagram principles, is used to *quantitatively* describe the oscillation inducing attributes of arbitrary signals.

Section 2.1 provides information about input shaping and vector diagrams that is relevant to this investigation. Section 2.2 demonstrates how arbitrary commands can be resolved into input shapers and unshaped commands. In Section 2.3, a case study is presented in which the oscillatory effects of rate-limited step commands on a harmonic oscillator are examined. Experimental results from an industrial bridge crane are used for validation.

2.1 Input Shaping and Vector Diagrams

Input shaping is used to generate commands that cause a system to cancel out its own motion-induced vibration. This was demonstrated by the experimental results shown previously in Figure 2.1(b). The velocity command shown in this figure, which moved the industrial crane without significant payload swing, was actually generated by convolving an

input shaper with the velocity command shown in Figure 2.1(a).

To gain a more formidable understanding of how input shapers reduce oscillation, consider the impulse response of an undamped second-order harmonic oscillator:

$$y_o(t) = \omega_n A_o \sin\Big(\omega_n(t - t_o)\Big) \cdot \mathbb{1}(t - t_o) \tag{2.1}$$

where A_o is the strength of the impulse, t_o is the time the impulse is applied, $\mathbb{1}(t)$ is the Heaviside function, and ω_n is the natural frequency of the second-order plant. The response from a sequence of impulses is a superposition of the response given in (2.1). After application of the n^{th} impulse, the steady-state response of the system can be described by:

$$y_{ss}(t) = \omega_n \sqrt{C^2 + S^2}\, \sin(\omega_n t - \Psi) \tag{2.2}$$

where,

$$C = \sum_{i=1}^{n} A_i \cos(\omega_n t_i), \quad S = \sum_{i=1}^{n} A_i \sin(\omega_n t_i), \quad \Psi = \operatorname{atan}\left(\frac{S}{C}\right)$$

In order to reduce the magnitude of steady-state vibration, the amplitudes, A_i, and time locations, t_i, of the impulse sequence can be determined by solving the set of constraint equations presented earlier in Section 1.2.1.

The steady-state amplitude of vibration induced into a linear system by an impulse sequence is *proportional* to the oscillation caused by a shaped command generated by convolving the impulse sequence with a baseline command (provided that the baseline command achieves a steady-state value) [16]. In this way, the amplitudes and time locations of the input shaper determine the vibration-inducing properties of the shaped command. The following section presents a vector diagram approach for calculating the amplitude of oscillation induced into a system by an impulse sequence.

2.1.1 Vector Diagrams

A vector diagram graphically represents an input shaper by using polar coordinates (r-θ space) in the phase plane [87, 78]. Figure 2.2 shows how a vector diagram is formed. The magnitude of each vector is created by setting r_i equal to the i^{th} amplitude, A_i, of an impulse sequence. The angle of each vector, θ_i, is equal to $\omega_n t_i$, where ω_n is the undamped

frequency of the system to which the input shaper is being applied, and t_i is the time of the i^{th} impulse.

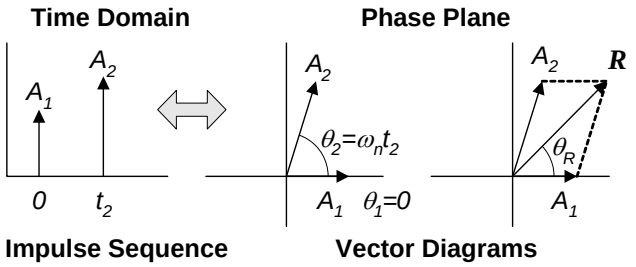


Figure 2.2: Input Shaper (Left); Equivalent Vector Diagram (Center); Resultant Vector (Right).

When a vector diagram is created in this manner, it becomes a useful tool for analyzing the oscillation-inducing properties of an input shaper. This is because the resultant vector sum, $\boldsymbol{R}$, is closely related to the steady-state impulse response in (2.2). This can be easily seen from the analytical expression for $\boldsymbol{R}$:

$$\boldsymbol{R} = \sqrt{R_x^2 + R_y^2}\,\angle \text{atan}\left(\frac{R_y}{R_x}\right) \tag{2.3}$$

where,

$$R_x = \sum_{i=1}^{n} A_i \cos(\omega_n t_i), \qquad R_y = \sum_{i=1}^{n} A_i \sin(\omega_n t_i)$$

Note that R_x and R_y are equal to C and S from (2.2), respectively. Furthermore, the magnitude of $\boldsymbol{R}$ is proportional to the steady-state amplitude in (2.2), which, in turn, is proportional to the oscillatory system response induced by a command shaped by the given impulse sequence. The angle of $\boldsymbol{R}$ is equal to the phase shift in (2.2). Therefore, $\boldsymbol{R}$ can be used as an indicator of anticipated system vibration in response to shaped commands.

Another important attribute of the resultant vector is that its corresponding impulse produces an oscillation in a system, equivalent to the steady-state oscillation caused by the original sequence of impulses. That is, the response of a second-order system to an impulse with the same magnitude and phase as $\boldsymbol{R}$ is identical to the response obtained from the original sequence of impulses after application of the last impulse.

2.1.2 Effects of Damping

When a harmonic oscillator has viscous damping, the vector diagram representation of vibration must be modified in two ways [87]. First, the damped natural frequency, ω_d, rather than the undamped frequency, ω_n, must be used to determine the angles of the impulse vectors. This corresponds to using:

$$\theta_i = \omega_d t_i \tag{2.4}$$

Second, the amplitudes of the phase-plane vectors must be scaled to account for the decay of the system response. Therefore, one should use a scaled version of the impulse amplitudes defined as[2]:

$$\widetilde{A}_i = A_i e^{-\zeta\omega_n(t_n - t_i)} \tag{2.5}$$

where ζ is the damping ratio, and t_n is the time of the last impulse. Using these modifications, the vector diagram approach to analyzing oscillatory properties of systems with input shaping may be applied generally to damped second-order systems.

2.2 Deconvolution Analysis of Arbitrary Signals

This section discusses how arbitrary reference commands can be interpreted as input-shaped signals. When the commands are interpreted in this way, input-shaping principles can be used to determine the vibration-inducing properties of these commands.

In order to interpret a signal as a shaped signal, it will be shown that a class of commands can always be resolved into an input shaper and a baseline command. In this way, an arbitrary signal, $x(t)$, can be expressed as the convolution of a baseline command, $w(t)$, and an input shaper, $IS(t)$.

Section 2.2.1 shows how this decomposition may be made when the input shaper is specified. Section 2.2.2 shows how the decomposition may be made when the baseline command is specified. Special attention is given to the resulting input shaper when the specified command is a step input. By analyzing this shaper with vector diagram techniques,

[2] An alternative scaling defined as $\widetilde{A}_i = A_i e^{\zeta\omega_n t_i}$ may be used. The latter produces a resultant vector that causes equivalent oscillation if applied at time $t = 0$. The former produces a resultant vector that causes equivalent oscillation when applied at time $t = t_n$.

it is shown that a simple relationship exists for determining the oscillatory effects of arbitrary signals on linear plants.

2.2.1 Specified Input Shaper

Any signal, $x(t)$, may be created from the convolution of a baseline command, $w(t)$, and an n-impulse input shaper having the form:

$$IS(t) = \sum_{i=1}^{n} A_i \delta(t - t_i) : \quad 0 \leq t_i < t_{i+1}, \quad A_i \neq 0 \tag{2.6}$$

where $\delta(t)$ is the Dirac delta function. If $w(t)$ is constrained to equal zero for $t < 0$, then the convolution relationship can be expressed as[3]:

$$x(t) = \int_0^{\infty} w(\tau) IS(t - \tau) d\tau \tag{2.7}$$

Given the form of the input shaper in (2.6), the expression for $x(t)$ may be simplified as:

$$x(t) = \sum_{i=1}^{n} A_i w(t - t_i) \tag{2.8}$$

Solving (2.8) for $w(t - t_1)$, and accounting for the t_1 time shift, $w(t)$ can be expressed in the form of a delay equation:

$$w(t) = \frac{1}{A_1} \left[x(t + t_1) - \sum_{i=2}^{n} A_i w(t + t_1 - t_i) \right] \tag{2.9}$$

The significance of (2.9) is that it verifies the existence of a baseline command, $w(t)$, for any shaper of the form described in (2.6), and the original signal, $x(t)$. The utility of (2.9) is that it provides a framework for a forward-time-marching numerical solution of $w(t)$, given $x(t)$ and the specified input shaper. This equation states that at time t, the value of $w(t)$ is proportional to the value of $x(t + t_1)$, and previous values of $w(t)$. Therefore, because $w(t) = 0$ for $t < 0$, the signal $x(t)$ and $IS(t)$ completely determine $w(t)$ for all $t \geq 0$.

The uniqueness of $w(t)$ is clearly demonstrated when (2.9) is used recursively to obtain a closed-form solution of the baseline command explicitly in terms of $x(t)$ and the input

[3] A corollary to the assumed form of $IS(t)$ and the constraint on $w(t)$ is that $x(t)$ will equal zero for $t < t_1$. This fact does not restrict $x(t)$ because the choice of t_1 is arbitrary and usually set equal to zero.

shaper parameters. The solution is valid for time $t < r(t_2 - t_1)$ where r is any positive integer:

$$w(t) = \sum_{p=0}^{r-1} \frac{(-1)^p}{A_1^{p+1}} \mathcal{E}(p)\mathcal{A}(p)\mathcal{X}(p,t) : \; t < r(t_2 - t_1) \tag{2.10}$$

where,

$$\mathcal{E}(p) = \overbrace{\sum_{i=2}^{n} \sum_{j=2}^{n} \cdots \sum_{q=2}^{n}}^{\text{multiplicity } p}$$

$$\mathcal{A}(p) = \overbrace{A_i A_j \cdots A_q}^{\text{multiplicity } p}$$

$$\mathcal{X}(p,t) = x(t + t_1(p+1) - \overbrace{t_i - t_j - \cdots - t_q}^{\text{multiplicity } p})$$

The expression $\mathcal{E}(p)\mathcal{A}(p)\mathcal{X}(p,t)$ is a compact notation for indicating the compound summation of impulse amplitudes multiplied by delayed values of $x(t)$. The number of compound summation signs, impulse amplitudes, and time delay terms is specified by p. Implementing (2.10) as a means of solving for $w(t)$ can be prohibitive because of its complexity. However, it does verify that a baseline command, $w(t)$, exists for any shaper of the form described in (2.6), and the original signal, $x(t)$.

Figure 2.3 depicts an example deconvolution of a rate-limited step command, $x(t)$ (solid line). The specified input shaper for this example is shown in the subplot. Using a forward-time-marching numerical solution with (2.9), $w(t)$ was found (dotted line).

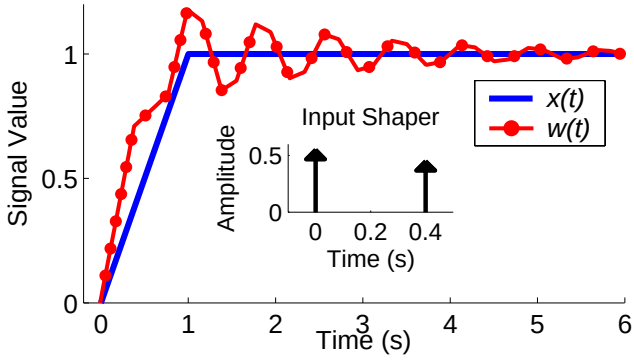


Figure 2.3: Deconvolution of *x(t)* into *w(t)* and a Two-Impulse Shaper.

The following summarizes the key points of this subsection:

Assumption: $w(t) = 0$ for $t < 0$.

Assumption: The specified input shaper has the form in (2.6).

Corollary: $x(t) = 0$ for $t < t_1$.

Conclusion: A baseline command, $w(t)$, exists that is completely determined by the signal $x(t)$ and the specified input shaper.

2.2.2 Specified Baseline Command

The general convolution relationship between $x(t)$, $w(t)$, and $IS(t)$ in the Laplace domain is:

$$IS(s) = \frac{X(s)}{W(s)} \tag{2.11}$$

If the baseline command, $w(t)$, is specified to be a unit step, and $x(t)$ is constrained to equal zero for $t \leq 0$, then the inverse Laplace transform of (2.11) yields:

$$IS(t) = \dot{x}(t) \tag{2.12}$$

If $x(t)$ is further constrained such that $x(t)$ is equal to a constant for $t \geq t_f$, then the time derivative of $x(t)$ can be expressed as an infinite sequence of impulses between time $t = 0$ and $t = t_f$:

$$\dot{x}(t) = \int_0^{t_f} \dot{x}(\tau)\delta(t-\tau)d\tau \tag{2.13}$$

The input shaper, $IS(t)$, is comprised of this sequence. Each impulse can be assigned an angle according to (2.4), and be scaled according to (2.5) to produce a vector:

$$\dot{\boldsymbol{x}}(t) = \overbrace{e^{-\zeta\omega_n(t_f-t)}}^{\text{scaling term}}\ \dot{x}(t)\quad \overbrace{e^{j\omega_d t}}^{\text{angle term}} \tag{2.14}$$

or, by substitution of (2.13), the vector relationship is expanded:

$$\dot{\boldsymbol{x}}(t) = \overbrace{e^{-\zeta\omega_n(t_f-t)}}^{\text{scaling term}} \int_0^{t_f} \dot{x}(\tau)\delta(t-\tau)d\tau \quad \overbrace{e^{j\omega_d t}}^{\text{angle term}} \tag{2.15}$$

The vector sum of this quantity may be obtained by summing the individual impulse vectors. After simplifying the summation, the resultant vector, $\boldsymbol{R}$, is[4]:

$$\boldsymbol{R} = e^{-\zeta\omega_n t_f} \int_0^{t_f} \dot{x}(t) e^{t(\omega_n\zeta+\omega_d j)}dt \tag{2.16}$$

This resultant vector has the same physical meaning and significance as the resultant vector in equation (2.3). It represents the geometric summation of impulse vectors in the phase plane. The magnitude and angle of $\boldsymbol{R}$ represent the oscillatory response of a harmonic system actuated by a signal derived from the convolution of $IS(t)$ with a unit step. More specifically, $|\boldsymbol{R}|$ may be interpreted as the ratio of two physical quantities: the oscillation amplitude induced by an arbitrary command, $x(t)$, and the oscillation amplitude induced by a unit step command. A useful form of (2.16) can be derived by examining $\dot{x}(t)$ in the Laplace domain:

$$\mathcal{L}[\dot{x}(t)] = \int_{-\infty}^{\infty} \dot{x}(t)e^{-st}dt \equiv sX(s) - x(0) \tag{2.17}$$

By applying the constraints that $x(t)$ is equal to zero for $t \leq 0$, and $x(t)$ is a constant for $t \geq t_f$, then evaluating (2.17) when s equals $-\zeta\omega_n - j\omega_d$, one obtains:

$$[sX(s)]_{s=-\zeta\omega_n - j\omega_d} = \int_0^{t_f} \dot{x}(t)e^{t(\omega_n\zeta + \omega_d j)}dt \tag{2.18}$$

which is precisely the integral portion of (2.16). Therefore,

$$\boldsymbol{R} = e^{-\zeta\omega_n t_f}\,[sX(s)]_{s=-\zeta\omega_n - j\omega_d} \tag{2.19}$$

Thus,

$$|\boldsymbol{R}| = e^{-\zeta\omega_n t_f}\,|sX(s)|_{s=-\zeta\omega_n - j\omega_d} \tag{2.20}$$

The significance of (2.16), (2.19), and (2.20) is that the magnitude of residual oscillation induced into a linear system by an arbitrary signal (limited by the aforementioned constraints) may be ascertained from either the integral of the signal's derivative, or the Laplace representation of the signal evaluated when s is set equal to the system poles.

This result is related to an analysis conducted by Bhat and Miu in the time domain [7], and later by Park et. al. in the digital domain [66]. They demonstrated that a system would exhibit zero residual vibration if the command signal had zeros at the system's flexible poles.

The following summarizes the key points of this subsection:

Assumption: $x(t) = 0$ for $t \leq 0$.

[4] A thorough analytical derivation of this expression, as well as alternative expressions for the $\boldsymbol{R}$-value are presented in Appendix A.

Assumption: $x(t) = C$ for $t \geq t_f$, $\quad C \in \mathbb{R}$.

Corollary: $\dot{x}(t) = 0$ for $t < 0$.

Corollary: $\dot{x}(t) = 0$ for $t > t_f$.

Conclusion: $\boldsymbol{R}$ represents the oscillatory response of a damped harmonic system to an arbitrary command, $x(t)$. $|\boldsymbol{R}|$ is proportional to the amplitude of the response. The phase of $\boldsymbol{R}$ is equal to the phase-shift of the response.

2.3 Case Study: Oscillatory Effects of a Rate-Limited Step Command

This section demonstrates the utility and efficacy of the described deconvolution analysis through an elementary case study. The simplicity of the example is purposeful in order that the *approach* to analyzing more complex systems is clear.

This example is concerned with the system illustrated in Figure 2.4. A rate-limited step command, $x(t)$, excites a damped harmonic oscillator, G, which responds with $y(t)$. The rising slew rate of the command is defined by the slope, m.

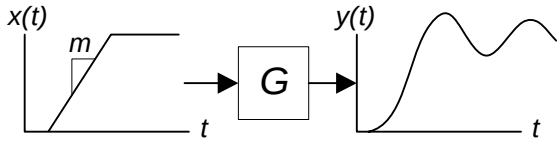


Figure 2.4: Response of a Harmonic Oscillator to a Rate-Limited Step Command.

The deconvolution analysis technique can determine the oscillatory effects of the rate-limited command on the plant by interpreting the signal, $x(t)$, as an input-shaped signal. Therefore, the rate-limited step command is considered here to be the product of an input shaper, $IS(t)$, convolved with a baseline command, $w(t)$:

$$x(t) = \overbrace{mt - (mt-1) \cdot \mathbb{1}\left(t - \frac{1}{m}\right)}^{\text{rate-limited step}} = w(t) * IS(t) \tag{2.21}$$

Section 2.3.1 addresses the problem when the baseline command is specified, but the input shaper is unknown. The resultant vector of the unknown shaper *quantitatively* describes the magnitude of residual oscillation. Section 2.3.2 interprets $x(t)$ when the input

shaper is specified, but the baseline command is unknown. The resulting baseline command *qualitatively* describes the residual oscillation. Section 2.3.3 compares experimentally measured residual oscillation with theoretical predictions.

2.3.1 Quantitative Oscillation Description

If a signal is deconvolved into a unit step and an unknown input shaper, then, according to (2.16), the resultant vector of the unknown shaper can be determined from an integration of $\dot{x}(t)$ multiplied by a scaling factor. Because this approach is easily implemented numerically, it is suitable for use with complicated forms of $x(t)$. For use with the simple signal of this example, (2.20) will be used. This equation determines the resultant vector of the unknown input shaper from the Laplace transform of $x(t)$:

$$X(s) = \frac{m}{s^2}\left[1 - e^{-s/m}\right] \tag{2.22}$$

The magnitude of the resultant vector is obtained by substituting (2.22) into (2.20) and simplifying to obtain:

$$|\boldsymbol{R}| = \frac{m}{\omega_n}\left[1 + e^{\frac{-2\zeta\omega_n}{m}} - 2e^{\frac{-\zeta\omega_n}{m}}\cos\left(\frac{\omega_d}{m}\right)\right]^{\frac{1}{2}} \tag{2.23}$$

In Figure 2.5, the magnitude of the resultant vector is plotted for several different damping ratios as a function of the slew rate limit, m. The horizontal axis has been normalized by a value of $m_c = \omega_n/2\pi$. The value $\omega_n/2\pi$ is significant; it is equal to the natural frequency (in Hz) of the oscillator, G. A large value of m is indicative of a fast rise time for the command, $x(t)$; as m approaches infinity, $x(t)$ approaches a step command.

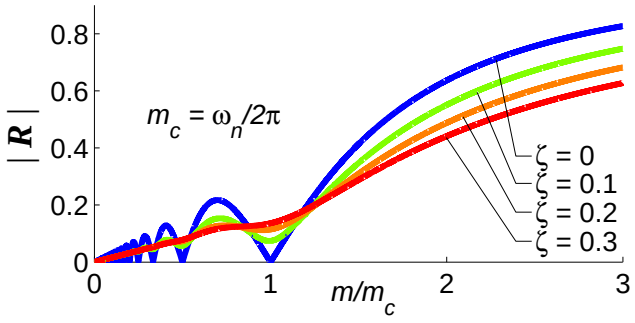


Figure 2.5: Resultant Vector Amplitudes for a Rate-Limited Step Command.

Because the amplitude of residual oscillation varies directly with the magnitude of $\boldsymbol{R}$, Figure 2.5 provides a concise quantitative description of the oscillatory effects of the rate-limited command on the harmonic oscillator. As the command becomes more aggressive, the magnitude of $\boldsymbol{R}$ asymptotically approaches 1. Therefore, a value of 1 on the vertical axis indicates that the system will respond to $x(t)$ with the same oscillation amplitude induced by a unit step command. Similarly, a value of 0.25 on the vertical axis indicates that the system will respond to $x(t)$ with a quarter of the oscillation amplitude induced by a unit step. When the $\boldsymbol{R}$-value is zero, $x(t)$ causes no residual oscillation.

Given this physical interpretation, an $\boldsymbol{R}$-value plot becomes a useful tool for concisely describing the oscillatory effects of arbitrary commands on a harmonic oscillator.

2.3.2 Qualitative Oscillation Description

An alternative to using the $\boldsymbol{R}$-value analysis for describing residual oscillation is to interpret an arbitrary signal as the convolution of a specified input shaper and an unknown baseline command. Then, the baseline command can be obtained by using (2.9).

The utility of this approach is that if the specified input shaper is correctly selected, then the behavior of the resulting baseline command is meaningful. It provides a qualitative description of oscillation insofar as it resembles the response of a plant to the arbitrary command. If the baseline command is oscillatory, then the system response to the arbitrary command will also be oscillatory. Conversely, if the baseline command quickly obtains a steady-state value, so will the system response.

This result is related to a fundamental property of input shaping. When a correctly designed shaper is convolved with a baseline command, the resulting shaped command produces zero residual oscillation in the plant after the baseline command has reached a *steady-state* value. If the baseline command is unceasingly transient, instead of obtaining a steady-state value, then the response of plant will also exhibit transient behavior.

To demonstrate the baseline command analysis, consider again the example of the rate-limited step. This command can be deconvolved into a baseline command and a two-impulse ZV shaper. Several of the baseline commands obtained from (2.9) are plotted with the solid

lines in Figure 2.6 for different values of m, when ζ equals 0.1. When convolved with the ZV shaper, these baseline commands exactly reproduce the original rate-limited commands. The response of G to the rate-limited step commands is shown with the dotted lines in Figure 2.6. Notice that the response signals closely resemble the baseline command signals in both amplitude and phase.

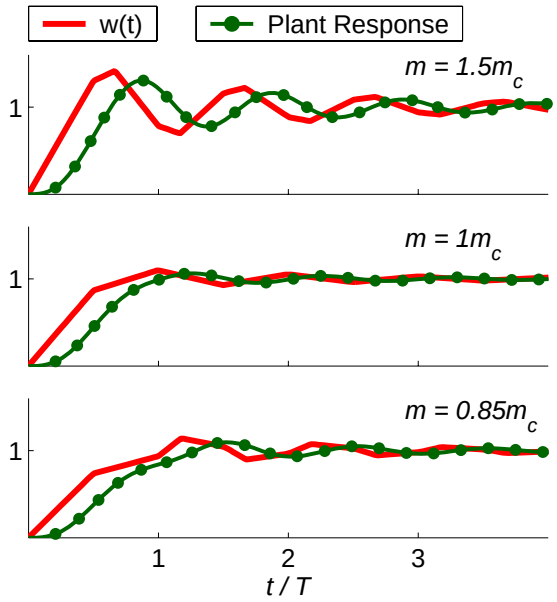


Figure 2.6: Baseline Commands Deconvolved from a Rate-Limited Step and a ZV Input Shaper (Solid). System Response to Rate-Limited Step Commands (Dotted).

Therefore, the utility of the baseline command analysis is that it yields a baseline command closely resembling the response of the given system to the given arbitrary command. If the analysis yields a baseline command that quickly obtains a steady-state value, then the plant will exhibit zero residual oscillation. Conversely, oscillatory baseline commands indicate an oscillatory plant response.

2.3.3 Experimental Verification

Figure 2.7 shows a 10-ton bridge crane. A linear model of the relationship between the velocity of the overhead trolley and the velocity of the suspended payload is [111]:

$$\frac{V_{out}}{V_{in}} = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{2.24}$$

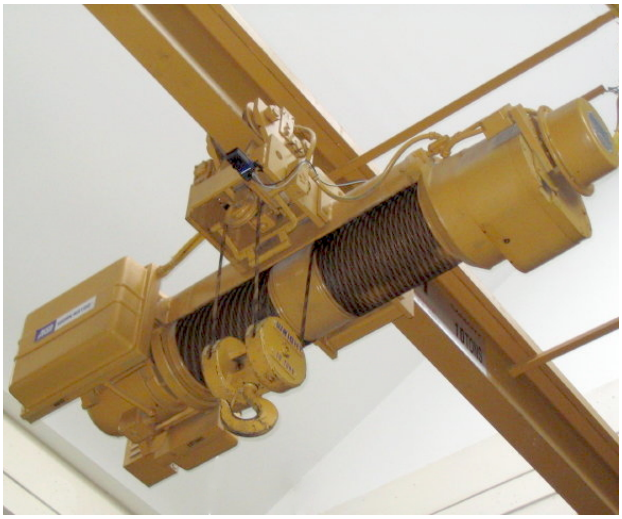


Figure 2.7: 10-Ton Bridge Crane.

The model reveals that the system behaves like a damped harmonic oscillator. The damping ratio is approximately 0.01 and the natural frequency is:

$$\omega_n = \sqrt{g/L} \tag{2.25}$$

where g is the acceleration due to gravity, and L is the length of the cable supporting the load.

A series of rate-limited step commands (in velocity) were issued to the crane. The slew-rate limit of the commands varied from $m = 0.7m_c$ to $m = 2.8m_c$. For each command, the residual oscillation amplitude of the suspended payload was measured. Each measured amplitude was divided by the oscillation amplitude occurring when a velocity step command was issued to the crane[5]. The resulting normalized amplitudes are plotted with the solid circles in Figure 2.8. The resultant vector plots from Figure 2.5 are superposed over the experimental results. This provides a comparison between the predicted and measured oscillation amplitudes. Given the low damping ratio of the crane, the measured amplitudes align closely with the predictions obtained using the $\boldsymbol{R}$-value analysis technique for the undamped case.

[5]A step in velocity is not realizable on the crane system because it requires infinite acceleration. Instead, the desired step command was approximated with a rate limited step command with $m \approx 10m_c$.

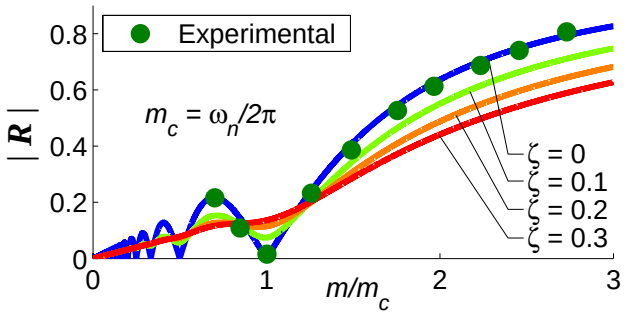


Figure 2.8: Experimental and Theoretical Resultant Vector Amplitudes for a Rate-Limited Step Command.

2.4 Conclusion

It is possible to deconvolve an arbitrary signal into both 1) a specified input shaper and baseline command, and 2) a specified baseline command and an input shaper. The utility of the deconvolution is that arbitrary signals can be analyzed from the perspective of input-shaped functions. This ability allows analytical tools, such as the vector diagram, to be utilized with many types of signals. By so doing, simple equations can be used to calculate the oscillatory effects of an arbitrary command on a harmonic oscillator. The efficacy of the deconvolution analysis technique was demonstrated by a case study in which the oscillatory response of rate-limited step commands on a harmonic oscillator was predicted. Experiments on a 10-ton industrial bridge crane confirmed theoretical results.

CHAPTER III

ANALYSIS OF SATURATION, RATE LIMITING, DEAD-ZONE, AND BACKLASH[1]

CHAPTER SUMMARY: *Actuator nonlinearities such as saturation, rate limiting, dead-zone, and backlash are frequently present in industrial machines. These hard nonlinearities can degrade the vibration-reducing properties of input-shaped signals. The* ***R****-value analysis tool provides a simple and informative means of quantifying the detrimental effects that hard nonlinearities have on input-shaped systems. This chapter presents an* ***R****-value analysis of saturation, rate limiting, dead-zone, and backlash on system using ZV, ZVD, and UMZV input shapers. Experiments performed on an industrial crane are used to verify theoretical developments.*

The ***R***-value analysis technique described in Chapter 2 provides a means for quantifying the oscillatory effects of arbitrary commands on a linear plant. For example, Figure 3.1 depicts the issuance of an arbitrary signal to a harmonic oscillator. The input, $y_{np}(t)$, conforms to the following constraints:

$$y_{np}(t) = 0, \qquad t \leq 0 \tag{3.1}$$

$$y_{np}(t) = H, \qquad t \geq t_f \tag{3.2}$$

The ***R***-value is a measure of both the phase and amplitude of the residual oscillation caused by $y_{np}(t)$. Residual oscillation is exhibited by a plant after an input has reached a steady-state value. The residual oscillation of G is shown with the dashed line in Figure 3.1, and

[1]Portions of this chapter are published in [98, 101, 5, 6, 100, 102].

can be quantitatively represented with the $\boldsymbol{R}$-value as:

$$|\boldsymbol{R}| = e^{-\zeta\omega_n t_f}|sY_{np}(s)|_{s=-\zeta\omega_n - j\omega_d} \tag{3.3}$$

The symbols ζ, ω_n, and ω_d are the damping ratio, natural frequency, and damped natural frequency of G, respectively.

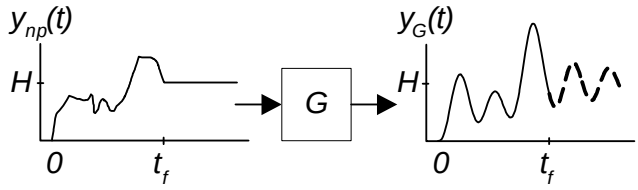
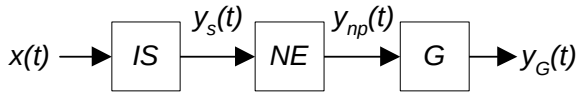


Figure 3.1: Input and Response of a Harmonic Oscillator to an Arbitrary Command.

Recall from the preceding chapter that the angle of $\boldsymbol{R}$ is equal to the phase shift of the residual oscillation, and the magnitude of $\boldsymbol{R}$ is directly proportional to the amplitude of residual oscillation at time t_f. More specifically, $|\boldsymbol{R}|$ is the ratio between two physical quantities: the oscillation amplitude, measured at time t_f, induced into G by the signal $y_{np}(t)$, and the oscillation amplitude, measured at time 0, that would be induced into G by a *unit* step command. For example, suppose that a given command yields $\boldsymbol{R} = 10$. This indicates that the amplitude of residual oscillation exhibited by the plant in response to the command would be 10-times greater than that induced by a unit step command. The $\boldsymbol{R}$-value of a ZV, ZVD, or UMZV-shaped command is 0 because no residual oscillation is induced.

By utilizing the capability to quantify oscillation caused by arbitrary commands, the detrimental effects of several actuator nonlinearities on systems using input shaping can be evaluated. To understand how this can be accomplished, consider Figure 3.2(a). This figure depicts a lightly-damped linear plant, G, the oscillation of which is to be suppressed by using an input shaper, IS. Ordinarily, when actuated by a ZV, ZVD, or UMZV-shaped command, G will exhibit *zero* residual oscillation. However, because the *nonlinear element*, NE, corrupts shaped commands before they are issued to G, the desired vibration suppression may not be achieved. Consequently, any residual oscillation exhibited by G can be attributed to the nonlinearity. Therefore, the effects of nonlinear elements on shaped

signals can be assessed quantitatively by determining the amplitude of residual oscillation exhibited by G in response to the corrupted signals. This is precisely the quantity that can be calculated when the corrupted signal, $y_{np}(t)$, is evaluated in terms of an $\boldsymbol{R}$-value.



(a) Serial Input Shaper, Nonlinear Element, and Linear Plant.

(b) Common Actuator Nonlinearities.

Figure 3.2: Input Shaping Implemented on a System with a Hard Nonlinearity.

Figure 3.2(b) depicts several common nonlinear elements that can reduce the vibration-suppression properties of shaped signals: saturation, rate limiting, dead-zone, and backlash. When assessing the effects of these elements, it is important to remember that the calculated $\boldsymbol{R}$-value depends not only on the given nonlinear element, but also on $x(t)$, the reference command issued to the system. The special case when this signal is a step command is significant. Step commands are common actuating signals. Furthermore, the oscillatory response of G when $x(t)$ is a step command can be regarded as a performance benchmark for the input-shaped system. This is analogous to the step-response performance benchmark often used for non-input-shaped systems. Thus, the general detrimental effects of saturation, rate limiting, dead-zone, and backlash on ZV, ZVD, and UMZV-shaped systems may be understood by considering $y_{np}(t)$ when $x(t)$ is a step.

In light of the preceding, the primary objective of this chapter is to:

- *Use the $\boldsymbol{R}$-value analysis technique to describe the effects that saturation, rate limiting, dead-zone, and backlash have on systems using ZV, ZVD, and UMZV input shapers.*

To accomplish this objective, the $\boldsymbol{R}$-value of $y_{np}(t)$ for the different shaper and nonlinear element combinations will be calculated for the case when $x(t)$ is a step command. Sections 3.1 through 3.4 present $\boldsymbol{R}$-value analyses of the aforementioned nonlinearities in

the context of the ZV, ZVD, and UMZV input shapers. The first analysis presented in Section 3.1 considerers rate limiting on systems using ZV input shapers. Considerable detail is included in this analysis in order to provide the reader not only with analysis results, but a working knowledge of the analysis procedure. However, in the remaining analyses, details are omitted, and only significant results are reported. Table 3.1 provides an overview of the different nonlinearities and input shapers that are considered, and the section where each analysis may be found.

Table 3.1: Section Location of Each Nonlinearity/Input Shaper $\boldsymbol{R}$-Value Analysis.

	Rate Limiting	**Dead-Zone**	**Saturation**	**Backlash**
ZV	Section 3.1.1	Section 3.2.1	Section 3.3.1	Section 3.4
ZVD	Section 3.1.2	Section 3.2.2	Section 3.3.2	Section 3.4
UMZV	Section 3.1.3	Section 3.2.3	Section 3.3.3	Section 3.4

Following the analysis of each input shaper and nonlinear element, results from experimental trials conducted on an industrial crane are presented. The results provide validation for predictions of system behavior derived from the various $\boldsymbol{R}$-value analyses.

3.1 R-Value Analysis of Rate Limiting

To quantify the oscillatory effects of rate limiting on ZV, ZVD, and UMZV input shapers, first consider the block diagram in Figure 3.3. This is similar to the block diagram shown in Figure 3.2(a), except the generic nonlinear element, NE, has been replaced with a rate limiter, characterized by the slew rate, m.

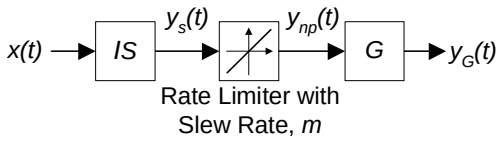
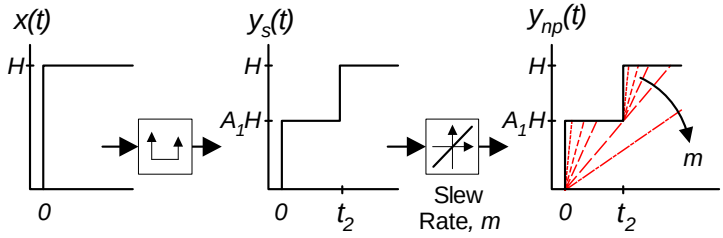


Figure 3.3: Serial Input Shaper, Rate Limiter, and Linear Plant.

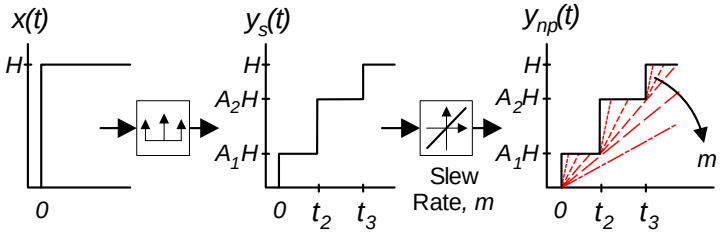
The rate limit places a bound on the maximum allowable slew rate of incoming signals. That is, the time derivative of signals exiting a rate limiter will be within the rate limit threshold range defined by $\pm m$. When m is very large, the allowable slew rate of incoming signals is also very large. As a result, $y_s(t)$ is modified very little. Conversely, when m is

very small, the maximum allowable slope of $y_s(t)$ is very small. In this case, $y_s(t)$ can be altered quite drastically. A thorough description of the functional relationship between the input and output of a rate limiting element is provided in Appendix B.

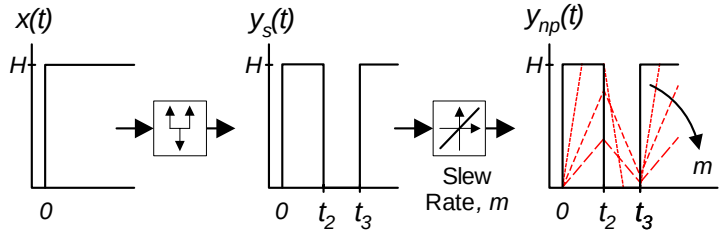
Figure 3.4 illustrates how ZV, ZVD, and UMZV-shaped step commands can be corrupted by a rate limiter. Note that for decreasing values of the rate limit, m, the shaped step commands are more severely corrupted. Both the ZV, and ZVD-shaped step commands have a characteristic "staircase" shape prior to being modified by the rate limit. However, after the rate limit, sufficiently low values of m reduce the staircase shape into a truncated ramp command. The UMZV-shaped step command exhibits more complicated modifications.



(a) ZV



(b) ZVD



(c) UMZV

Figure 3.4: Shaped Step Commands Corrupted by Rate Limiting.

3.1.1 ZV-Shaper

A piecewise formula for a ZV-shaped step command of magnitude H, corrupted by a rate limiter is:

$$y_{np}(t) = \begin{cases} m\Big[t - (t-t_a)\mathbb{1}(t-t_a) + (t-t_2)\mathbb{1}(t-t_2) - (t-t_b)\mathbb{1}(t-t_b)\Big], & m \geq \frac{A_1 H}{t_2} \\ m\Big[t - (t-t_c)\mathbb{1}(t-t_c)\Big] & , \; m < \frac{A_1 H}{t_2} \end{cases} \tag{3.4}$$

where $\mathbb{1}(t)$ is the Heaviside function, t_2 is the time of the second impulse for the ZV shaper, and t_a through t_c are defined as:

$$t_a = A_1 H/m, \quad t_b = (A_2 H/m) + t_2, \quad t_c = H/m$$

By forming the Laplace transform of (3.4), and then substituting the result into (3.3), one obtains the $\boldsymbol{R}$-value representation of the oscillatory effects of rate limiting on ZV input shapers:

$$\left|\boldsymbol{R}\right| = \begin{cases} e^{-\zeta\omega_n t_b}\left|\frac{m}{Q}\left(1 - e^{t_a Q} + e^{t_2 Q} - e^{t_b Q}\right)\right|, & m \geq \frac{A_1 H}{t_2} \\ e^{-\zeta\omega_n t_c}\left|\frac{m}{Q}\left(1 - e^{t_c Q}\right)\right| & , \; m < \frac{A_1 H}{t_2} \end{cases} \tag{3.5}$$

where,

$$Q = \zeta\omega_n + j\omega_d \tag{3.6}$$

Equation (3.5) is plotted in Figure 3.5 for different damping ratios as a function of the rate limit, m. The vertical axis has been normalized by H. The horizontal axis has been normalized by $m_c = H/T$, where T is the period of oscillation for the system. The critical rate limit, m_c is significant. It represents the constant acceleration required to accelerate the system from rest to a speed of H in the time equal to one period of oscillation.

Because the amplitude of residual oscillation varies directly with $|\boldsymbol{R}|$, non-zero values of the lines indicate when the rate-limiting nonlinearity diminishes the vibration-reducing properties of the ZV-shaped command. As the rate limit, m, approaches infinity, the lines asymptotically approach zero, indicating that an infinite rate limit will not induce residual oscillation into the system. This result was anticipated because as m approaches very high values, the rate limit does not corrupt the shaped command, and the system behaves

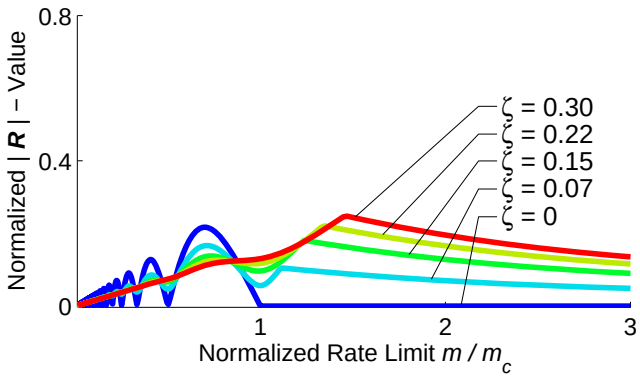


Figure 3.5: Oscillatory Effects of Rate Limiting on ZV-Shaped Step Commands.

as if the nonlinear element was not present. As m is decreased, the shaped command is corrupted, producing commands that cause oscillation.

One feature of Figure 3.5 is particulary noteworthy. Notice that for the case of zero damping, the $\boldsymbol{R}$-value is identically equal to zero when m is greater than m_c. This result indicates that over this range of rate limits, the nonlinearity has no detrimental effect on the vibration reducing properties of ZV-shaped step commands.

3.1.1.1 Comparison of $\boldsymbol{R}$-Value Plots with and without Input Shaping

Figure 3.5 clearly reveals that rate limiting can significantly reduce the vibration suppression properties of ZV-shaped signals. However, the primary objective of utilizing input shaping is to decrease oscillation. Therefore, a natural question arising from this circumstance is:

> *Does the oscillation exhibited by the input-shaped and rate-limited system exceed the oscillation that would occur if the input-shaping filter were not used?*

To address this question, consider the block diagram shown in Figure 3.6, which represents the control scenario when input shaping is not used. In this circumstance, the step input is directly modified by the rate limiter, which actuates the plant with a rate-limited step command. The oscillatory effects of the rate limiter when input shaping is not used can be determined from an $\boldsymbol{R}$-value analysis of the rate limited step command. The time-domain representation of the rate limited step is:

$$y_{np}(t) = mt - \mathbb{1}\left(t - \frac{H}{m}\right) \cdot m\left(t - \frac{H}{m}\right) \tag{3.7}$$

Substituting the Laplace transform of (3.7) into (3.3) yields the $\boldsymbol{R}$-value representation of rate limiting when input shaping is not used:

$$|\boldsymbol{R}| = e^{-\zeta\omega_n H/m}\left|\frac{m}{Q}\left(1 - e^{\frac{H}{m}Q}\right)\right| \tag{3.8}$$

where Q was defined previously in (3.6).

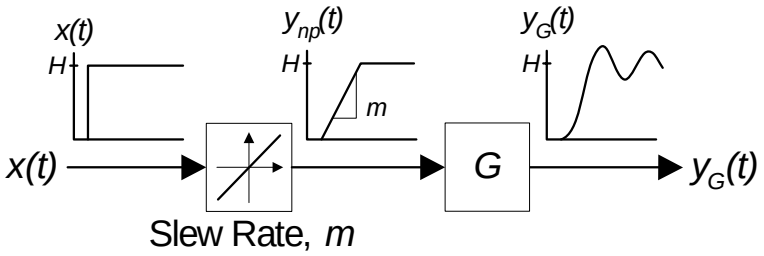


Figure 3.6: Rate Limited System without Input Shaping.

Equation (3.8) is plotted with the dashed lines in Figure 3.7 for several different damping ratios. The horizontal and vertical axes have been normalized by m_c and H, respectively. For comparison purposes, the previous results of Figure 3.5 are superposed on the graph with the solid lines. For low values of m, the dashed curves coincide with the solid curves. At these m-values, the rate limit is so severe that the characteristic "staircase" shape of the ZV-shaped step command is effaced, and therefore is identical to a rate-limited step. As m is increased, the dashed curves asymptotically approach the value of one.

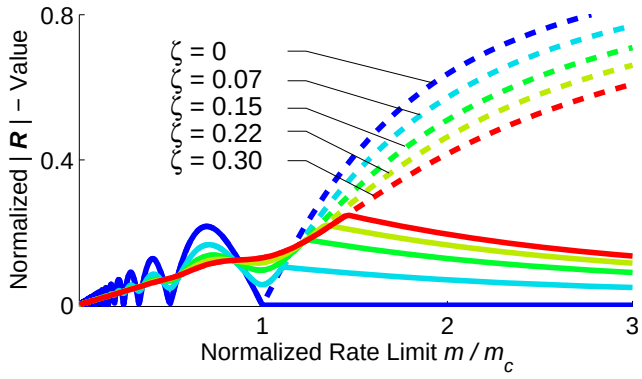


Figure 3.7: Oscillatory Effects of Rate Limiting on ZV-Shaped and Unshaped Step Commands.

Therefore, the answer to the previously posed question is succinctly answered by examination of Figure 3.7. Although rate limiting does reduce the efficacy of the ZV shaper,

input shaping still reduces oscillatory system behavior to levels at, or below, those obtained without input shaping. Clearly, for values of $m > m_c$, the input-shaped signals induce much less oscillation than the unshaped commands.

The $\boldsymbol{R}$-value plots from both the shaped and unshaped systems provide a concise description of how rate limiting affects shaped systems. The parameter range over which rate limiting inhibits effective input shaping is clearly revealed. The magnitude of oscillation caused by the corrupted step and corrupted ZV-shaped step commands is also revealed. Additionally, the $\boldsymbol{R}$-value comparison between the shaped and unshaped systems provide a means to evaluate the effectiveness of input shaping at reducing oscillation in the presence of rate limiting.

3.1.1.2 Analysis Procedure

The procedure used to determine the oscillatory effects of rate limiting is summarized:

1. Graphically represent how a shaped step command is corrupted by the nonlinearity;
2. Determine the time domain function of the corrupted command;
3. Obtain the oscillatory effects of the corrupted command in terms of an $\boldsymbol{R}$-value;
4. Plot the $\boldsymbol{R}$-value function;
5. Plot the $\boldsymbol{R}$-value function for the case when input shaping is not used;
6. Determine the oscillatory effects of the nonlinearity by comparing the two $\boldsymbol{R}$-value plots.

Identical procedures have been carried out to determine the detrimental effects of rate limiting on the remaining input shapers.

3.1.2 ZVD-Shaper

Figure 3.4(b) illustrated how rate limiting corrupts ZVD-shaped step commands. The uncorrupted command is characterized by a three-step "staircase" shape. The corrupted command can be grouped into four characteristic shapes representatively shown in Figure 3.8.

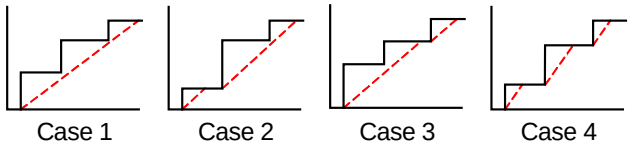


Figure 3.8: Characteristic Corruption of ZVD-Shaped Step Command by a Rate Limit.

The $\boldsymbol{R}$-value representation of the corrupted commands is:

$$|\boldsymbol{R}| = \begin{cases} e^{-\zeta\omega_n t_d}\left|\frac{m}{Q}\left(1 - e^{t_d Q}\right)\right| & , \text{ Case 1} \\ e^{-\zeta\omega_n t_e}\left|\frac{m}{Q}\left(1 - e^{t_a Q} + e^{t_2 Q} - e^{t_e Q}\right)\right| & , \text{ Case 2} \\ e^{-\zeta\omega_n t_c}\left|\frac{m}{Q}\left(1 - e^{t_g Q} + e^{t_3 Q} - e^{t_c Q}\right)\right| & , \text{ Case 3} \\ e^{-\zeta\omega_n t_c}\left|\frac{m}{Q}\left(1 - e^{t_a Q} + e^{t_2 Q} - e^{t_b Q} + e^{t_3 Q} - e^{t_c Q}\right)\right|, & \text{ Case 4} \end{cases} \tag{3.9}$$

where Q was previously defined in (3.6), t_2 and t_3 are the impulse times for the ZVD shaper, and t_a through t_g are defined as:

$$\begin{aligned} t_a &= A_1 H/m & t_b &= t_2 + A_2 H/m \\ t_c &= t_3 + A_3 H/m & t_d &= H/m \\ t_e &= t_2 + (A_2 + A_3)H/m & t_g &= (A_1 + A_2)H/m \end{aligned}$$

To facilitate a description of when each of the four cases occur, the following three slopes and critical damping ratio are defined:

$$\begin{aligned} M_1 &= A_1 H/t_2 \\ M_2 &= A_2 H/(t_3 - t_2) \\ M_3 &= (A_1 + A_2)H/t_3 \\ \zeta_c &= \frac{\ln(2)}{\sqrt{\ln(2)^2 + \pi^2}} \cong 0.215 \end{aligned}$$

Case selection is determined based upon the value of the linear plant's damping ratio, ζ, and the value of the rate limit, m, in relation to M_1, M_2, and M_3, according to the case selection summary in Table 3.2.

Equation (3.9) is plotted with the solid lines in Figure 3.9 for different damping ratios as a function of the rate limit. The $\boldsymbol{R}$-value plots for unshaped steps are shown by the dashed lines. The horizontal and vertical axes are again normalized by m_c and H, respectively.

This figure exhibits similar characteristics as the $\boldsymbol{R}$-value plot for the ZV case. For low values of m, the shaped and unshaped $\boldsymbol{R}$-value plots coincide. As m is increased, the shaped

Table 3.2: Case Selection Table for the ZVD Shaper.

	$m \le M_1$	$M_1 < m \le M_2$	$M_2 < m$
$0 \le \zeta \le \zeta_c$	Case 1	Case 2	Case 4
$\zeta_c < \zeta$	Case 1	Case 3	Case 4
	$m \le M_3$	$M_3 < m \le M_1$	$M_1 < m$

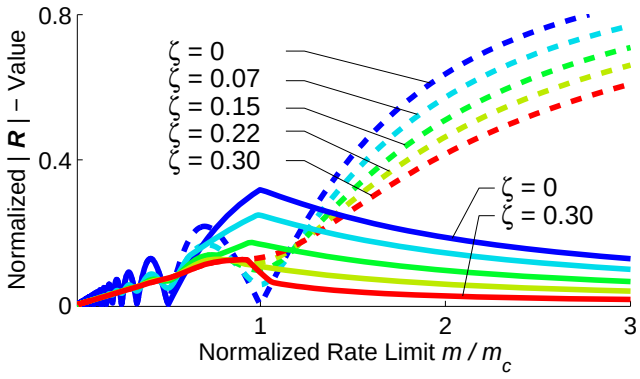


Figure 3.9: Oscillatory Effects of Rate Limiting on ZVD-Shaped and Unshaped Step Commands.

commands produce decreasing levels of vibration, indicated by the asymptotic approach of the solid lines to zero, while the unshaped commands cause increasing levels of vibration.

This figure concisely reveals the slew rate parameter range that causes ZVD-shaped commands to be less effective at reducing oscillation. Generally, for values of $m < m_c$, shaped and unshaped commands will produce equivalent levels of oscillation. For higher values of m, it is clear that shaped commands result in much less oscillation than the unshaped counterparts.

One significant distinction exists between the $\boldsymbol{R}$-value plots for the ZVD shaper and the $\boldsymbol{R}$-value plots for the ZV shaper. Figure 3.9 shows that for values of $m > m_c$, lightly damped systems will exhibit higher levels of residual vibration than more heavily damped systems. This scenario is reversed for the ZV case shown previously in Figure 3.5. These results suggest that, for lightly damped systems, ZV-input shapers are less sensitive to the oscillation-inducing effects of rate limiting than ZVD-input shapers.

This result is significant because it represents a departure from the conventional wisdom of the input shaping community. Generally, ZVD shapers are regarded as being superior

to ZV shapers insofar as their capacity to reduce vibration on systems with uncertain parameters. While this notion is correct for frequency and damping uncertainty, the same cannot be stated about rate limiting. Therefore, in selecting an appropriate input shaper for a given system, the robustness of the ZVD shaper to frequency and damping uncertainty should be weighed against the detrimental effects of rate limiting.

3.1.3 UMZV-Shaper

A UMZV-shaped step command, corrupted by rate limiting, can be grouped into three characteristic shapes depicted in Figure 3.10.

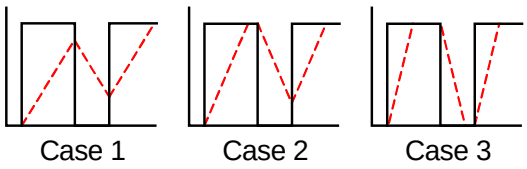


Figure 3.10: Characteristic Corruption of UMZV-Shaped Step Command by a Rate Limit.

The $\boldsymbol{R}$-value representation of the corrupted commands is:

$$\left|\boldsymbol{R}\right| = \begin{cases} e^{-\zeta\omega_n t_d}\left|\frac{m}{Q}\left(1-2e^{t_2Q}+2e^{t_3Q}-e^{t_dQ}\right)\right| & , \text{ Case 1} \\ e^{-\zeta\omega_n t_e}\left|\frac{m}{Q}\left(1-e^{t_aQ}-e^{t_2Q}+2e^{t_3Q}-e^{t_eQ}\right)\right| & , \text{ Case 2} \\ e^{-\zeta\omega_n t_c}\left|\frac{m}{Q}\left(1-e^{t_aQ}-e^{t_2Q}+e^{t_bQ}+e^{t_3Q}-e^{t_cQ}\right)\right|, & \text{ Case 3} \end{cases} \tag{3.10}$$

where Q was previously defined in (3.6), t_2 and t_3 are the impulse times for the UMZV shaper, and t_a through t_e are defined as:

$$\begin{aligned} t_a &= H/m & t_b &= t_2 + H/m \\ t_c &= t_3 + H/m & t_d &= 2(t_3 - t_2) + H/m \\ t_e &= 2t_3 - t_2 \end{aligned}$$

To facilitate a description of when each of the three cases occur, the following slopes are defined:

$$M_1 = H/t_2 \qquad M_2 = H/(t_3 - t_2)$$

Case selection is determined based on the value of the rate limit, m, in relation to M_1, and M_2, according to the case selection summary in Table 3.3.

Table 3.3: Case Selection Table for the UMZV Shaper.

$m \leq M_1$	$M_1 < m \leq M_2$	$M_2 < m$
Case 1	Case 2	Case 3

Equation (3.10) is plotted with the solid lines in Figure 3.11 for different damping ratios as a function of the rate limit. The dashed lines in Figure 3.11 represent the $\boldsymbol{R}$-value when input shaping is not used. Two features of this figure are noteworthy. First, it is apparent that rate limiting can be more detrimental to the vibration suppression properties of the UMZV shaper than the ZV or ZVD shapers. This is attested to by the fact that the plots for the UMZV case attain higher $\boldsymbol{R}$-values. Second, the detrimental effects of rate limiting on UMZV shapers subside at higher values of m than for either the ZV or ZVD shapers. These results suggest that for a moderately sluggish system with $m_c < m < 6m_c$, a ZV or ZVD shaper will yield superior vibration reduction than a UMZV shaper.

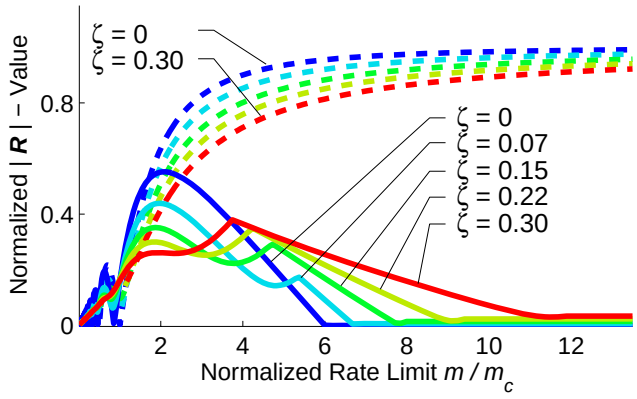


Figure 3.11: Oscillatory Effects of Rate Limiting on UMZV-Shaped and Unshaped Step Commands.

3.1.4 Experimental Verification of Detrimental Effects

Figure 3.12 shows a photograph of a 10-ton industrial bridge crane. A block diagram of the crane actuation process is shown in Figure 3.13.

The block, NP, represents the nonlinear functionality of the system drives and motors. The drives and motors accept velocity commands issued to the crane, and convert the reference velocity to the actual velocity of the overhead trolley. A programmable rate limit,

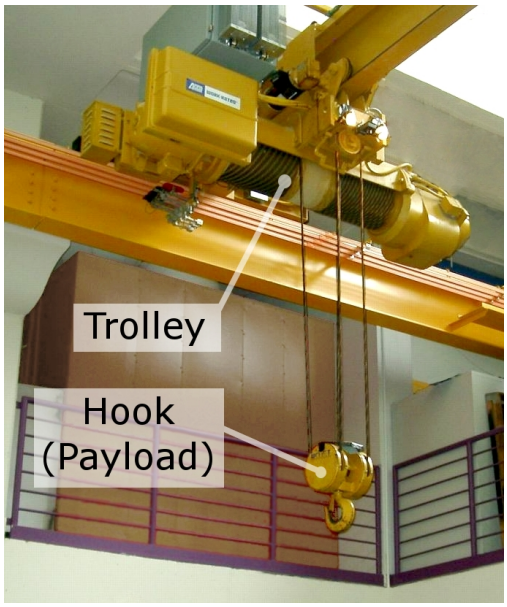


Figure 3.12: 10-Ton Bridge Crane.

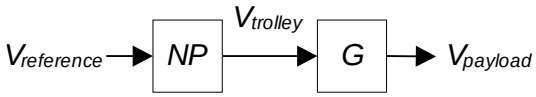


Figure 3.13: Crane Actuation Block Diagram.

dead-zone, and saturator are associated with the system drives. The block, G, is a linear transfer function relating the velocity of the overhead trolley to the velocity of the suspended payload [111]:

$$G = \frac{V_{payload}}{V_{trolley}} = \frac{2\zeta\omega_n s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{3.11}$$

This model reveals that the linear portion of the system behaves like a damped harmonic oscillator. The damping ratio is approximately 0.01.

The results of Sections 3.1.1 through 3.1.3 predict that if ZV, ZVD, or UMZV-shaped commands are issued to the crane, and significant rate limiting is exhibited by the drives, then the vibration suppression properties of the shaped commands will be degraded. As a result, oscillation will be induced into the payload.

To experimentally validate the oscillatory effects of rate limiting, the programmable drives were configured to exhibit an acceleration limit bounded by $\pm m$. The parameters associated with dead-zone and saturation were set such that the effects from these elements

were negligibly small. Several ZV, ZVD, and UMZV-shaped velocity step commands were then issued to the crane. For each trial, the rate limit, m, was set at a constant value ranging from very sluggish ($m \approx 0.25m_c$) to very rapid ($m \approx 10m_c$). The residual payload swing caused by each velocity command was measured. The command and response from a typical trial are shown in Figure 3.14. During this trial, a ZVD input shaper was used. Notice that the velocity response of the trolley to the reference velocity exhibits noticeable rate limiting.

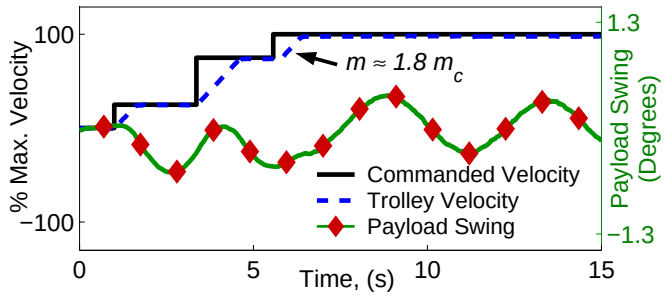
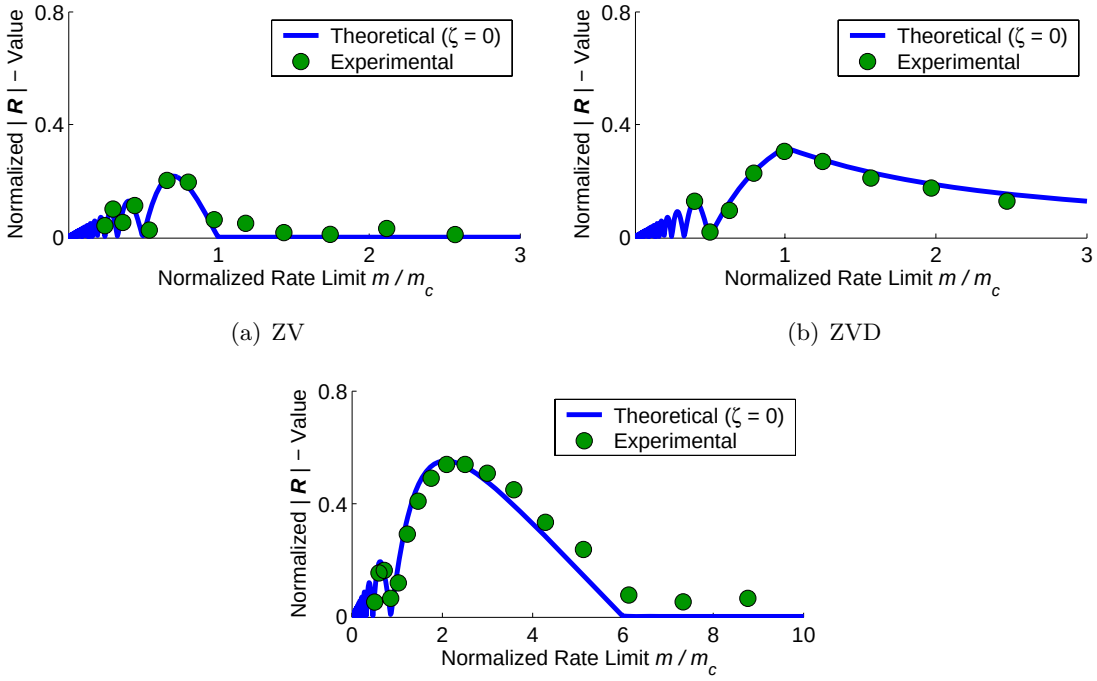


Figure 3.14: ZVD-Shaped Velocity Command and System Response for $m \approx 1.8m_c$.

In order to compare the experimental results with the theoretical $\boldsymbol{R}$-values presented previously, the measured data must be normalized by the payload swing caused by a velocity step command. To obtain this value, a single velocity step command was issued to the system with the rate limit parameter set at its maximum ($m \approx 23m_c$). At this maximum rate limit, the crane's oscillatory response closely mimics the response that would theoretically be attained in response to a true step in velocity. The residual payload oscillation in response to this command was measured, and used for normalization. The resulting ratios are equal to the $\boldsymbol{R}$-value for a lightly damped system with the given value of m. They represent the experimentally obtained oscillatory effects of rate limiting on ZV, ZVD, and UMZV-shaped commands. The results obtained from this procedure are plotted with the circles in Figures 3.15(a) through 3.15(c), for the ZV, ZVD, and UMZV shapers, respectively.

For comparison purposes, the theoretical $\boldsymbol{R}$-value curves for undamped motion are also plotted. The similarity between the measured and theoretical prediction validates that 1) rate limiting reduces the effectiveness of ZV, ZVD, and UMZV-shaped commands in a significant and predictable manner, and 2) the $\boldsymbol{R}$-value analysis technique is a reliable tool

for predicting such detrimental effects.



(a) ZV

(b) ZVD

(c) UMZV

Figure 3.15: Experimental and Theoretical Effects of Rate Limiting on ZV, ZVD, and UMZV-Shaped Step Commands.

3.2 R-Value Analysis of Dead-Zone

Consider the block diagram in Figure 3.16. This is similar to the block diagram shown previously in Figure 3.2(a), except that the generic nonlinear element, NE, has been replaced with a dead-zone. The dead-zone element generates zero output in response to input signals that are within a region called the dead-zone. The upper and lower limits of the dead-zone are specified by $\pm d$. A thorough description of this behavior is provided in Appendix B.

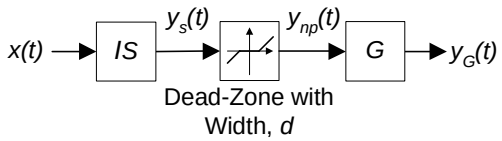
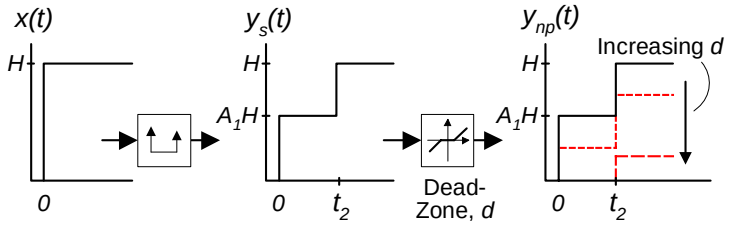


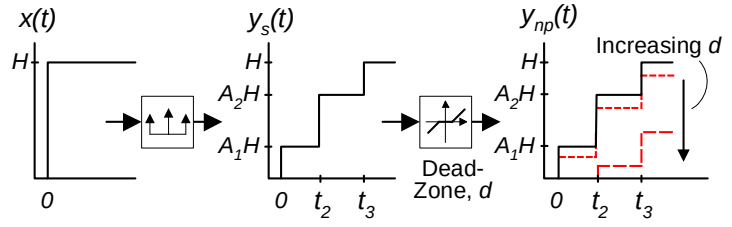
Figure 3.16: Serial Input Shaper, Dead-Zone, and Linear Plant.

Figure 3.17 illustrates how ZV, ZVD, and UMZV-shaped step commands can be corrupted by a dead-zone. For both the ZV and ZVD cases, the dead-zone truncates the

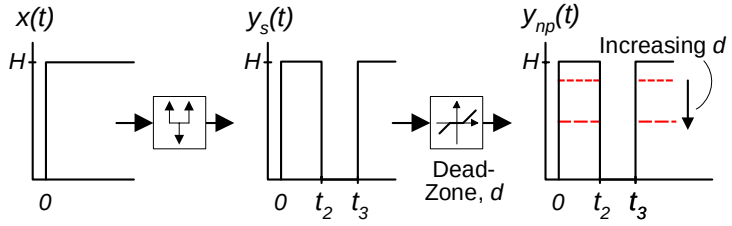
characteristic staircase features of the shaped step commands. For large dead-zone widths, the number of staircase features is reduced. The effect of the dead-zone on the UMZV-shaped step command is less severe in that the overall shape and number of features in the command are maintained; only the magnitude of the command is affected.



(a) ZV



(b) ZVD



(c) UMZV

Figure 3.17: Shaped Step Commands Corrupted by Dead-Zone.

3.2.1 ZV-Shaper

The oscillatory influence that dead-zone has on ZV shaping can be determined from the $\boldsymbol{R}$-value representation of a ZV-shaped step command corrupted by dead-zone. When the magnitude of the step command is H, this representation is:

$$|\boldsymbol{R}| = \begin{cases} e^{-\zeta\omega_n t_2}\left|A_1 H - d + A_2 H e^{t_2 Q}\right|, & 0 \le d < A_1 H \\ e^{-\zeta\omega_n t_2}\left|(H-d)e^{t_2 Q}\right| & , \; A_1 H \le d < H \\ 0 & , \; H \le d \end{cases} \tag{3.12}$$

The parameters A_1, A_2, and t_2 are the impulse amplitudes and times associated with the ZV input shaper. The parameter, Q, was previously defined in (3.6). Equation (3.12) can be simplified by utilizing the expressions for the input shaper parameters in (1.9) and the expression for Q in (3.6). After simplification, (3.12) can be expressed as:

$$|\boldsymbol{R}| = \begin{cases} de^{-\zeta\omega_n t_2}, & 0 \leq d < A_1 H \\ H - d \quad , & A_1 H \leq d < H \\ 0 \qquad\quad , & H \leq d \end{cases} \tag{3.13}$$

Equation 3.13 is plotted with the solid lines in Figure 3.18 for different damping ratios as a function of the dead-zone width. Both the vertical and horizontal axes have been normalized by H. For comparison purposes, the $\boldsymbol{R}$-value plot for the case when input shaping is not used is shown with the dashed line.

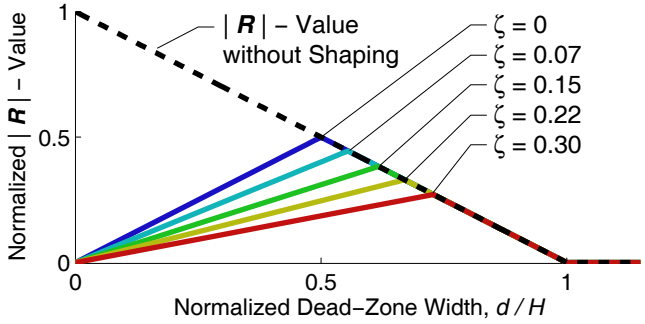


Figure 3.18: Oscillatory Effects of Dead-Zone on ZV-Shaped and Unshaped Step Commands.

Because the amplitude of residual system oscillation varies directly with the magnitude of $\boldsymbol{R}$, non-zero values of the solid lines indicate when the nonlinearity diminishes the vibration-reducing properties of the ZV-shaped commands. The shaped system exhibits the maximum amount of residual oscillation when the plant has zero damping, and the value of the dead-zone, d, is half the value of H. When the dead-zone width is equal to zero, the $\boldsymbol{R}$-value for the shaped case is identically zero. This observation reveals the anticipated result that in the absence of dead-zone, the harmonic oscillator will not exhibit residual oscillation in response to ZV-shaped commands. Likewise, for values of d greater than H, $\boldsymbol{R}$ is also identically zero. However, this is because the system is not actuated and remains stationary.

The comparison of the shaped $\boldsymbol{R}$-value plots with the unshaped $\boldsymbol{R}$-value plot provides a baseline by which to evaluate the effectiveness of ZV-shaping at reducing oscillation in the presence of dead-zone. Clearly, for values of d less than $H/2$, it is predicted that the input-shaped signals induce much less oscillation than the unshaped counterparts.

3.2.2 ZVD-Shaper

The $\boldsymbol{R}$-value representation of a ZVD-shaped step command corrupted by dead-zone is:

$$|\boldsymbol{R}| = \begin{cases} e^{-\zeta\omega_n t_3}\left|A_1 H - d + A_2 H e^{t_2 Q} + A_3 H e^{t_3 Q}\right|, & 0 \le d < A_1 H \\ e^{-\zeta\omega_n t_2}\left|A_1 H + A_2 H - d + A_3 H e^{t_2 Q}\right| & , \ A_1 H \le d < (A_1 + A_2) H \\ e^{-\zeta\omega_n t_3}\left|(H-d) e^{t_3 Q}\right| & , \ (A_1 + A_2) H \le d < H \\ 0 & , \ H \le d \end{cases} \tag{3.14}$$

The parameters A_{1-3} and t_{2-3} are the impulse amplitudes and times associated with the ZVD input shaper. Q was defined previously in (3.6). After simplification, (3.14) can be expressed as:

$$|\boldsymbol{R}| = \begin{cases} d e^{-\zeta\omega_n t_3} & , \ 0 \le d < A_1 H \\ \left|d e^{-\zeta\omega_n t_2} - H A_2/2 - H A_3\right|, & A_1 H \le d < (A_1 + A_2) H \\ H - d & , \ (A_1 + A_2) H \le d < H \\ 0 & , \ H \le d \end{cases} \tag{3.15}$$

The influence that dead-zone has on ZVD-shaped systems can be ascertained from Figure 3.19 where (3.15) is plotted with the solid lines as a function of the dead-zone width for several different damping ratios. Like the previous figure, the horizontal and vertical axes have been normalized by H. For comparison purposes, the $\boldsymbol{R}$-value plot for the case when input shaping is not used is shown with the dashed line.

Two features of this figure are significant. First the solid curves remain below an $\boldsymbol{R}$-value of 0.25 throughout the dead-zone parameter range. In contrast, the corresponding plots for the case of ZV input shaping in Figure 3.18 can reach twice this value. The second significant feature is related to the range of dead-zone widths over which shaped $\boldsymbol{R}$-value curves remain below the unshaped $\boldsymbol{R}$-value curve. This favorable range for the ZVD input shaper ($0 \le d < 0.75H$) is larger than the range for the ZV input shaper ($0 \le d < 0.5H$).

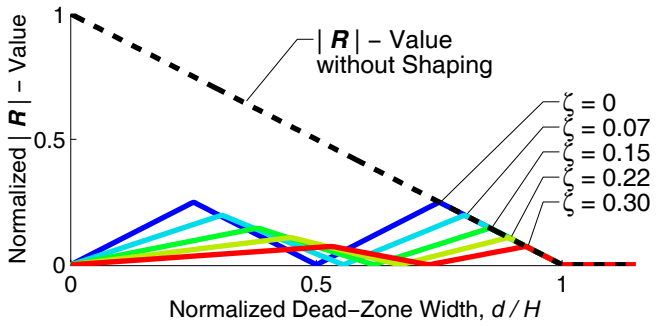


Figure 3.19: Oscillatory Effects of Dead-Zone on ZVD-Shaped and Unshaped Step Commands.

By considering the features of Figure 3.19 in light of Figure 3.18, one may conclude that, generally speaking, ZVD shapers are less susceptible to the oscillatory effects of dead-zone than ZV shapers. Therefore, a ZVD-shaped system with dead-zone will tend to exhibit less oscillation over a wider range of dead-zone widths than a similar system using a ZV input shaper.

3.2.3 UMZV-Shaper

The modification of a UMZV-shaped step command by a dead-zone element was previously illustrated in Figure 3.17(c). The modified commands have the same shape as the UMZV-shaped step command, but are attenuated by an amount equal to the dead-zone width. Accordingly, identical commands could be produced if an equally attenuated step command was filtered by a UMZV input shaper. The significance of this fact is that the dead-zone element has no effect on the vibration suppression properties of a UMZV-shaped step command. This is because the corrupted command is, itself, a UMZV-shaped step command.

To demonstrate this fact, consider the time-domain representation of a UMZV-shaped step command modified by a dead-zone element:

$$y_{np}(t) = (H-d)\mathbb{1}(t) - (H-d)\mathbb{1}(t-t_2) + (H-d)\mathbb{1}(t-t_3) \tag{3.16}$$

where t_2 and t_3 are the impulse times for the UMZV shaper. The Laplace transform of this equation is:

$$Y_{np}(s) = \frac{H-d}{s}\left[1 - e^{-st_2} + e^{-st_3}\right] \tag{3.17}$$

By substituting (3.17) into (3.3) and simplifying, one obtains the $\boldsymbol{R}$-value representation of the UMZV-shaped step command modified by a dead-zone element:

$$|\boldsymbol{R}| = e^{-\zeta\omega_n t_3} \left|(H-d)\left(1 - e^{t_2 Q} + e^{t_3 Q}\right)\right| \tag{3.18}$$

This representation of the $\boldsymbol{R}$-value may be expanded by incorporating the expression for Q from (3.6), and evaluating the absolute value term. After simplification, one obtains:

$$|\boldsymbol{R}| = (H-d)e^{-\zeta\omega_n t_3}\sqrt{C^2 + S^2} \tag{3.19}$$

where,

$$C = \sum_{i=1}^{3}(-1)^{(i+1)} e^{\zeta\omega_n t_i}\cos(\omega_d t_i) \tag{3.20}$$

$$S = \sum_{i=1}^{3}(-1)^{(i+1)} e^{\zeta\omega_n t_i}\sin(\omega_d t_i) \tag{3.21}$$

Equations (3.20) and (3.21) are identically equal to the vibration constraint equations, (1.6) and (1.7), when the constraint equations are evaluated for the case of the UMZV input shaper. Therefore, (3.20) and (3.21) are equal to zero. Consequently, (3.19) is also equal to zero, which implies that the dead-zone element does not reduce the vibration suppression properties of the UMZV-shaped step command.

Although dead-zone has no effect on the vibration reducing properties of UMZV-shaped step commands, this may not be the case for arbitrary commands that are filtered by the UMZV shaper. This is because the $\boldsymbol{R}$-value analysis is only an indicator of general system performance. It is used in much the same way as the step response from a conventionally analyzed system. Both analyses provide performance benchmarks for general system behavior. Therefore, while the previous analysis does not guarantee the non-effect of dead-zone on all UMZV-shaped signals, it does indicate that dead-zone affected systems will exhibit relatively low levels of residual oscillation over the entire range of dead-zone widths.

3.2.4 Experimental Verification of Detrimental Effects

Sections 3.2.1 and 3.2.2 predict that if ZV or ZVD-shaped commands are issued to a linear plant, and the plant actuators exhibit significant dead-zone behavior, then the vibration

suppression properties of the shaped commands will be degraded. The crane previously described in Section 3.1.4 was used to verify these predictions. Recall that the behavior of the payload in response to crane motion mimics the behavior of a lightly damped second-order system, and the actuating drives and motors exhibit dead-zone, saturation, and rate limiting. The extent to which the drives and motors exhibit the nonlinear behaviors can be varied by changing the programmable parameters associated with each element.

To experimentally validate the oscillatory effects of dead-zone, it is desirable to isolate the oscillation induced by dead-zone from the oscillation caused by either saturation or rate limiting. To accomplish this, the drives were configured to exhibit a dead-zone width specified by $\pm d$; the saturation threshold was set sufficiently high to ensure that issued commands would not saturate the motors; and the rate-limiting parameter was set at its maximum value of approximately $28m_c$ (m_c was previously defined in Section 3.1.1). Ideally, the rate limit threshold would be set infinitely high. However, this would correspond to an infinite acceleration of the crane. Because physical systems cannot instantaneously change velocity, the infinite rate limit is unrealizable. Therefore, the oscillatory effects of dead-zone cannot be entirely isolated from the oscillatory effects of rate limiting. Nevertheless, at the set value of $28m_c$, the effects from rate limiting are negligibly small. An analysis conducted to verify the minimal influence of rate limiting is presented in Appendix C.

After configuration of the drives, several ZV and ZVD-shaped velocity step commands were issued to the crane. For each command, the dead-zone width, d, was set at a constant value ranging from zero to H (H is equal to the maximum velocity of the crane). The residual payload swing caused by the velocity command was measured. The system motion for a typical trial is shown in Figure 3.20. This plot shows the commanded velocity, trolley velocity, and payload swing for the trial when d was set to 25% of the maximum velocity, and a ZV input shaper was used. The residual swing amplitude corresponding to each tested d-value was measured. These measurements were subsequently normalized by the swing amplitude induced by a velocity step command of magnitude H.

The resulting ratios obtained from this set of experiments are plotted with the circles in Figure 3.21(a) and Figure 3.21(b) for the ZV and ZVD input shapers, respectively.

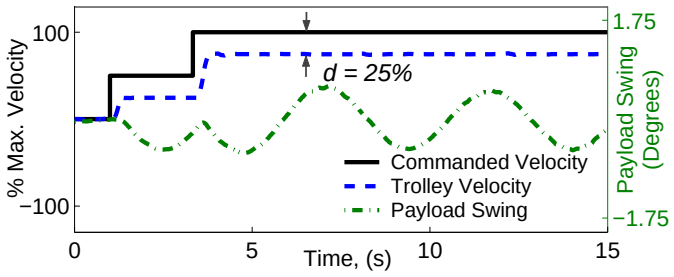
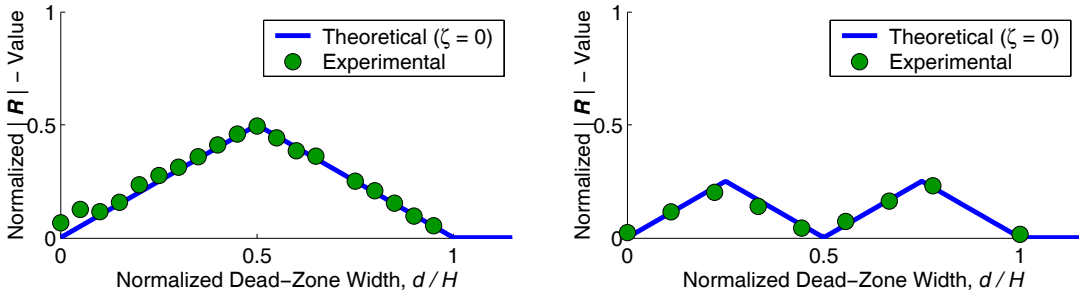


Figure 3.20: ZV-Shaped Velocity Command and System Response for $d = 25\%$.

For comparison purposes, the theoretical $\boldsymbol{R}$-value curves for undamped motion are also plotted. The similarity between the theoretical and experimentally determined $\boldsymbol{R}$-values validates that 1) dead-zone reduces the effectiveness of ZV and ZVD-shaped commands in a significant and predictable manner, and 2) the $\boldsymbol{R}$-value analysis technique is a reliable tool for predicting such detrimental effects.



(a) ZV (b) ZVD

Figure 3.21: Experimental and Theoretical Effects of Dead-Zone on ZV and ZVD-Shaped Step Commands.

3.3 R-Value Analysis of Saturation

Figure 3.22 depicts a block diagram of an input-shaped system with a saturator nonlinearity. The saturation block truncates values of the shaped signal, $y_s(t)$, that are outside the saturation threshold range defined by $\pm p$. For values of p that are larger than the maximum value of $y_s(t)$, the shaped command is not affected at all. As p decreases, the maximum value of $y_{np}(t)$ also decreases. A more thorough description of saturation is provided in Appendix B.

Figure 3.23 illustrates how ZV, ZVD, and UMZV-shaped step commands are modified

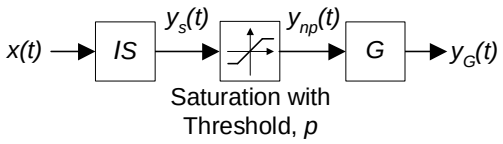
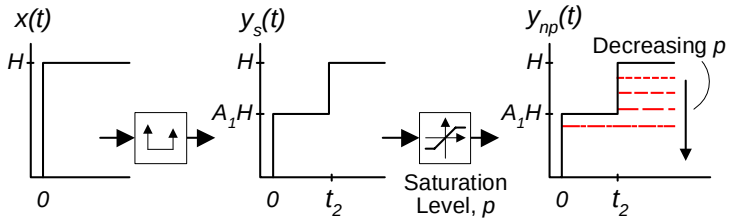
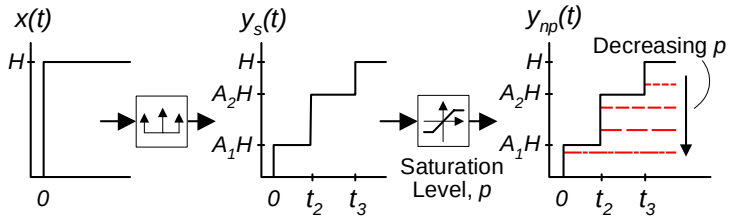


Figure 3.22: Serial Input Shaper, Saturator, and Linear Plant.

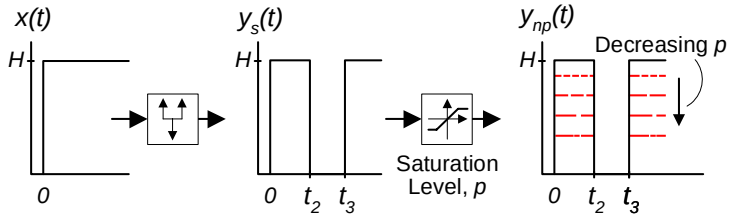
by a saturator. As the saturation threshold is lowered, the shaped commands are more severely modified. For the ZV case, the characteristic two-step staircase command becomes a one-step staircase command. Likewise, for the ZVD case, the characteristic three-step staircase is reduced to a two-step, and then a one-step staircase command as the saturation limit decreases. The effect of saturation on the UMZV-shaped step command is identical to the effect that dead-zone has on the UMZV-shaped step command. The overall shape and number of features in the command are maintained; only the magnitude of the command is affected.



(a) ZV



(b) ZVD



(c) UMZV

Figure 3.23: Shaped Step Commands Corrupted by Saturation.

3.3.1 ZV-Shaper

The oscillatory influence that saturation has on ZV shaping can be determined from the $\boldsymbol{R}$-value representation of a ZV-shaped step command corrupted by saturation. If the magnitude of the step command is H, then this representation is[2]:

$$|\boldsymbol{R}| = \begin{cases} pe^{-\zeta\omega_n t_2} & , \ 0 \leq p < A_1 H \\ e^{-\zeta\omega_n t_2} \left| A_1 H + (p - A_1 H) e^{t_2 Q} \right| & , \ A_1 H \leq p < H \\ 0 & , \ H \leq p \end{cases} \tag{3.22}$$

where A_1 and t_2 are the impulse amplitude and time associated with the ZV shaper. Q was defined previously in (3.6). Simplification of (3.22) produces:

$$|\boldsymbol{R}| = \begin{cases} pe^{-\zeta\omega_n t_2}, & 0 \leq p < A_1 H \\ H - p & , \ A_1 H \leq p < H \\ 0 & , \ H \leq p \end{cases} \tag{3.23}$$

Note that (3.23) is identical to (3.13), the corresponding $\boldsymbol{R}$-value equation for dead-zone, except that the dead-zone parameter, d, is replaced with the saturation threshold, p. This indicates that even though saturation and dead-zone modify ZV-shaped step commands differently, the different commands produce equivalent levels of residual vibration.

The oscillatory equivalence of saturation to dead-zone is also evident in the $\boldsymbol{R}$-value plot for saturation, shown with the solid lines in Figure 3.24. These lines are identical to the solid lines in Figure 3.18, the $\boldsymbol{R}$-value plot for dead-zone.

The dashed line in Figure 3.24 represents the $\boldsymbol{R}$-value plot for the case when input shaping is not used. Like the previous analyses revealed, shaped commands perform as well, or better than, non-input-shaped commands at reducing oscillation, even in the presence of saturation.

3.3.2 ZVD-Shaper

The oscillatory influence that saturation has on ZVD shaping can be determined from the $\boldsymbol{R}$-value representation of a ZVD-shaped step command corrupted by saturation. If the

[2]This expression represents the oscillation amplitude of G at time t_2, in response to the ZV-shaped step command, corrupted by saturation.

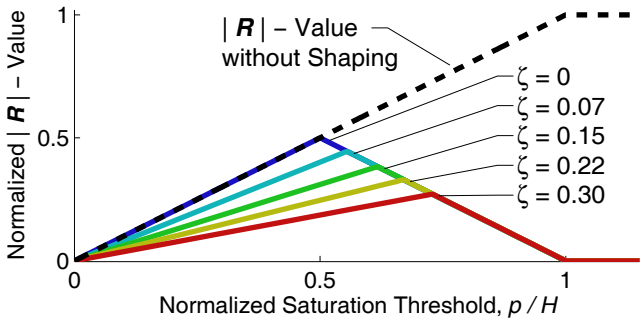


Figure 3.24: Oscillatory Effects of Saturation on ZV-Shaped and Unshaped Step Commands.

magnitude of the step command is H, then this representation is[3]:

$$|\boldsymbol{R}| = \begin{cases} pe^{-\zeta\omega_n t_3} & , \ 0 \le p < A_1 H \\ e^{-\zeta\omega_n t_3}\left|A_1 H + (p - A_1 H)e^{t_2 Q}\right| & , \ A_1 H \le p < (A_1 + A_2)H \\ e^{-\zeta\omega_n t_3}\left|A_1 H + A_2 H e^{t_2 Q} + (p - A_1 H - A_2 H)e^{t_3 Q}\right| & , \ (A_1 + A_2)H \le p < H \\ 0 & , \ H \le p \end{cases} \tag{3.24}$$

Again, the parameters A_i and t_i are associated with the ZVD input shaper, and Q was defined previously in (3.6). After simplification, (3.24) can be expressed as:

$$|\boldsymbol{R}| = \begin{cases} pe^{-\zeta\omega_n t_3} & , \ 0 \le p < A_1 H \\ \left|pe^{-\zeta\omega_n t_2} - HA_2/2 - HA_3)\right| & , \ A_1 H \le p < (A_1 + A_2)H \\ H - p & , \ (A_1 + A_2)H \le p < H \\ 0 & , \ H \le p \end{cases} \tag{3.25}$$

Equation (3.25) is identical to (3.15), the corresponding $\boldsymbol{R}$-value equation for dead-zone, except that the dead-zone parameter, d, is replaced with the saturation threshold, p. This indicates that even though saturation and dead-zone modify ZVD-shaped step commands differently, the different commands produce equivalent amplitudes of residual oscillation. This result is similar to that obtained for the case of ZV input shaping.

These results are confirmed by Figure 3.25, where (3.25) is plotted with the solid lines.

[3]This expression represents the oscillation amplitude of G at time t_3, in response to the ZVD-shaped step command, corrupted by saturation.

This $\boldsymbol{R}$-value representation of saturation is identical to the $\boldsymbol{R}$-value representation of dead-zone.

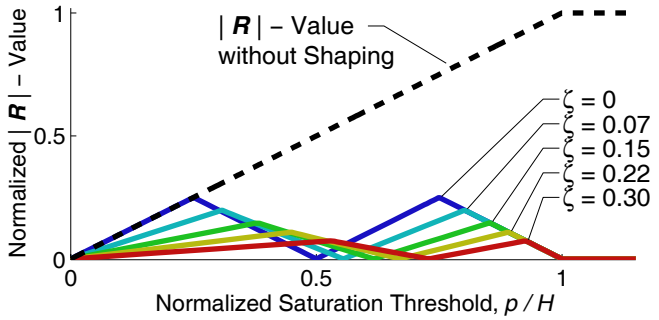


Figure 3.25: Oscillatory Effects of Saturation on ZVD-Shaped and Unshaped Step Commands.

The dashed line in Figure 3.25 represents the $\boldsymbol{R}$-value plot for the case when input shaping is not used. Again, it can be observed that shaped commands perform as well or better than non-input-shaped commands at reducing oscillation, even in the presence of saturation. Additionally, considering Figure 3.25 in light of Figure 3.24 reveals that the ZVD shaper is generally less affected by saturation than the ZV shaper.

3.3.3 UMZV-Shaper

As stated previously, saturation and dead-zone modify UMZV-shaped step commands in equivalent ways. The shape and number of features in the modified command are maintained; only the magnitude of the command is affected. Consequently, UMZV-shaped step commands modified by saturation could also be produced by filtering an appropriately scaled step command with a UMZV shaper. An $\boldsymbol{R}$-value analysis of this type of command for the case of dead-zone was presented in Section 3.2.3, and could easily be applied to the case of saturation. Therefore, it can be concluded that saturation does not degrade the vibration-suppressing properties of UMZV-shaped step commands.

Although saturation has no effect on the vibration-reducing properties of UMZV-shaped step commands, this may not be the case for arbitrary commands that are filtered by the UMZV shaper. As stated previously, the $\boldsymbol{R}$-value analysis is only an indicator of general system performance, analogous to the step response of a conventionally analyzed system.

Therefore, the non-effect of saturation on the vibration suppression properties of UMZV-shaped step commands indicates that, for the general cause of UMZV shaping, saturation will cause relatively low levels of residual oscillation over the entire range of saturation thresholds.

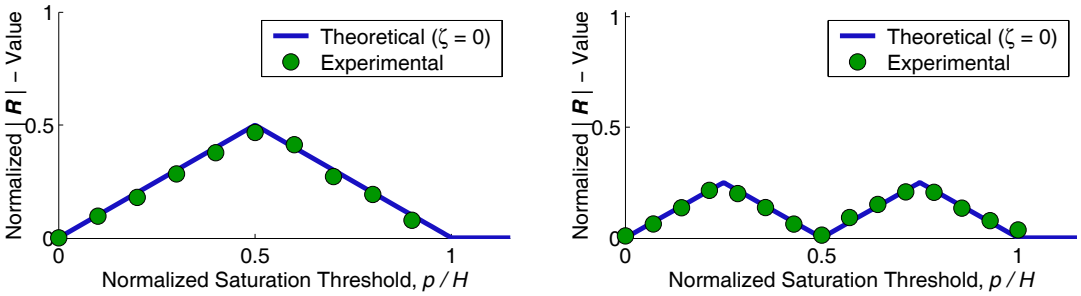
3.3.4 Experimental Verification of Detrimental Effects

Sections 3.3.1 and 3.3.2 predict that if ZV or ZVD-shaped commands are issued to a linear plant, and the plant actuators saturate, then the vibration suppression properties of the shaped commands may be degraded. The same crane used to test the oscillatory effects of rate limiting and dead-zone was used to verify these predictions. To accomplish this, the drives were configured to saturate at a threshold p; the dead-zone width was set to zero; and the rate limiting effects were minimized by setting the rate limit parameter to its maximum value. As discussed in Section 3.2.4, the oscillatory effects of rate limiting cannot be entirely eliminated from the experiment. However, Appendix C shows that when the rate limit parameter is set at its maximum value, the oscillatory effects caused by rate limiting are negligibly small.

After configuration of the drives, several ZV and ZVD-shaped velocity step commands were issued to the crane. For each command, the saturation threshold, p, was set at a constant value ranging from zero to H (H is equal to the maximum velocity of the crane). The residual payload swing caused by the velocity command was measured. The residual swing amplitude corresponding to each of the p-values tested was measured. These measurements were subsequently normalized by the swing amplitude induced by a velocity step command of magnitude H.

The resulting ratios obtained from this set of experiments are plotted with the circles in Figure 3.26(a) and Figure 3.26(b) for the ZV and ZVD input shapers, respectively. For comparison purposes, the theoretical $\boldsymbol{R}$-value curves for undamped motion are also plotted. The similarity between the theoretical and experimentally determined $\boldsymbol{R}$-values validates that 1) saturation reduces the effectiveness of ZV and ZVD-shaped commands in a significant and predictable manner, and 2) the $\boldsymbol{R}$-value analysis technique is a reliable

tool for predicting such detrimental effects.



(a) ZV (b) ZVD

Figure 3.26: Experimental and Theoretical Effects of Saturation on ZV and ZVD-Shaped Step Commands.

3.4 R-Value Analysis of Backlash

Figure 3.27 depicts a block diagram of an input-shaped system with a backlash nonlinearity. The backlash nonlinearity is an element in which a change in input, $y_s(t)$, causes an equal change in the output, $y_{np}(t)$. However, when the input changes direction, an initial change in input has no effect on the output. This amount of decoupled movement is referred to as the backlash.

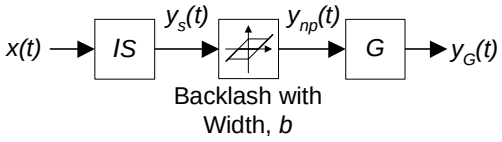


Figure 3.27: Serial Input Shaper, Backlash, and Linear Plant.

A backlash element is often used to model the "play" between two components of a mechanical system. A simple example of a system with backlash is illustrated in Figure 3.28. In the configuration shown, the input must travel a distance b, before the output responds. A more thorough description of backlash is provided in Appendix B.

The functional relationship between the input and output of a backlash element and the input and output of a dead-zone element are identical, with the exception of a single attribute: A backlash element is active every time the input changes direction; a dead-zone element is active only when the input is near the value of zero. Because of the similarities

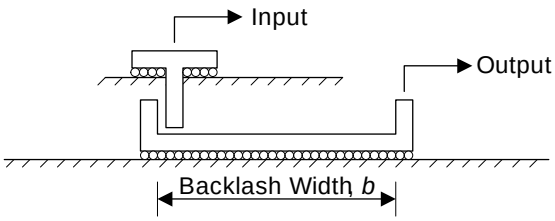
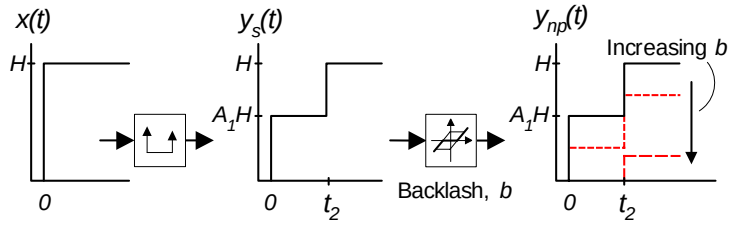
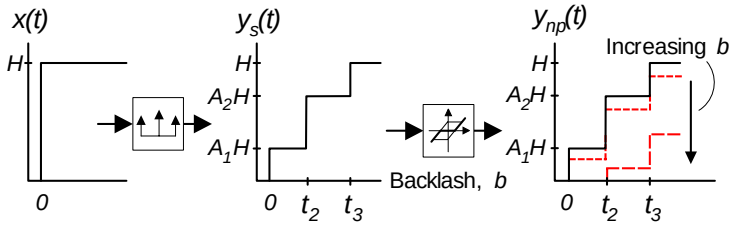


Figure 3.28: Mechanical System with Backlash.

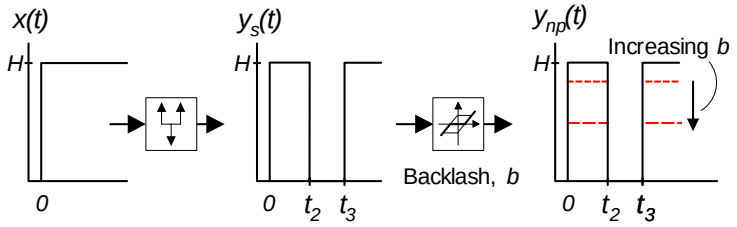
between backlash and dead-zone, the way that each of these elements modify ZV, ZVD, and UMZV-shaped step commands is identical. This is evident by examining Figure 3.29. The modified commands in this figure are identical to the modified commands for the case of dead-zone previously shown in Figure 3.17.



(a) ZV



(b) ZVD



(c) UMZV

Figure 3.29: Shaped Step Commands Corrupted by Backlash.

Given the indistinguishable way that backlash and dead-zone modify ZV, ZVD, and UMZV-shaped step commands, it may be concluded that the oscillatory effects of backlash are the same as the oscillatory effects of dead-zone. Therefore, the $\boldsymbol{R}$-value equations for

backlash, and the corresponding $\boldsymbol{R}$-value plots are the same as those shown previously in Section 3.2, with the dead-zone width, d, replaced by the backlash width, b.

3.5 Conclusion

Rate limiting, dead-zone, saturation, and backlash inherent in actuating drives and motors can undesirably modify input-shaped reference commands. By using the $\boldsymbol{R}$-value analysis technique, these detrimental effects were quantified for the special case when step commands actuate a given system. The results obtained from these analyses provide baselines by which to assess more general system behavior, such as the behavior resulting from arbitrary commands. Several results obtained from the analyses are noteworthy: 1) although the various nonlinear elements inhibit complete vibration suppression, the shaped systems still reduced oscillation to levels at, or below, the levels obtained without input shaping; 2) for lightly damped systems, the robust ZVD shaper is more sensitive to the effects of rate limiting than the non-robust ZV shaper; 3) when modified by dead-zone, saturation, or backlash, ZVD-shaped commands generally produce less oscillation than ZV-shaped commands; 4) UMZV shapers are the least susceptible to the detrimental effects of dead-zone, saturation, and backlash; and 5) the UMZV input shaper is the least affected by high rate limits, and the most affected by moderate rate limits. These results were verified experimentally on an industrial crane.

CHAPTER IV

MITIGATION OF SATURATION, RATE LIMITING, DEAD-ZONE, AND BACKLASH[1]

CHAPTER SUMMARY: *Actuator nonlinearities such as saturation, rate limiting, dead-zone, and backlash are frequently present in industrial machines. These behaviors can degrade the vibration-reducing properties of input-shaped signals. Previously, the oscillatory effects that these elements have on common input shapers were quantified. This chapter presents strategies to mitigate these detrimental effects when the value of the saturation, rate limiting, dead-zone or backlash parameter can be estimated. The robustness of the mitigation techniques to parameter uncertainty are evaluated. Experiments performed on an industrial crane are used to verify the effectiveness of the mitigation techniques and predictions of robustness.*

The detrimental effects of saturation, rate limiting, dead-zone, and backlash on shaped systems were evaluated in Chapter 3. It was shown that these nonlinear elements can significantly reduce the vibration suppression properties of shaped signals. As a result, input-shaped systems with actuator nonlinearities can exhibit unwanted oscillation. Although it was shown that these levels of oscillation are generally less severe than oscillation caused without input shaping, it still may be desirable to counteract these effects. Consequently, the objective of this chapter is to:

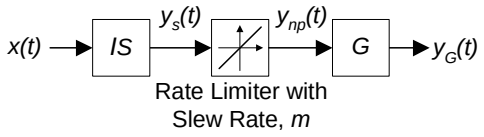
- *Present mitigation techniques that eliminate or reduce the oscillation inducing effects of saturation, rate limiting, dead-zone, and backlash on systems using input-shaping filters.*

[1]Portions of this chapter are published in [98, 101, 100, 102].

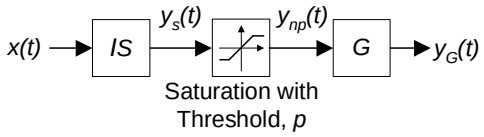
The techniques used to mitigate the oscillatory effects of saturation and rate limiting are similar to one another, and are presented in Section 4.1. The techniques used for dead-zone and backlash are also similar to one another; these are presented in Section 4.2.

4.1 Mitigating Saturation and Rate Limiting through Input Signal Conditioning

To understand how oscillation caused by saturation or rate limiting can be mitigated, consider the generic input-shaping block diagrams shown in Figure 4.1. These block diagrams depict a shaped signal, $y_s(t)$, that is modified by either a rate limiter (4.1(a)) or saturator (4.1(b)). The modified command actuates a harmonic oscillator, G, with natural frequency, damped natural frequency, and damping ratio of ω_n, ω_d, and ζ, respectively.



(a) Rate Limiting.



(b) Saturation.

Figure 4.1: Serial Input Shaper, Nonlinear Element, and Linear Plant.

One strategy for mitigating the detrimental effects of a rate limit is to generate input-shaped commands that do not exceed the rate limiting bounds defined by $\pm m$. The *rate limit mitigation condition* is expressed as:

$$|\dot{y}_s(t)| \leq m \tag{4.1}$$

If (4.1) is satisfied, then the shaped signal, $y_s(t)$, is unaffected by the rate limit. Thus, $y_{np}(t) = y_s(t)$, and a pure input-shaped command actuates the plant.

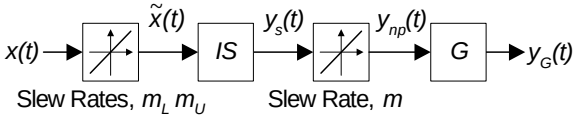
A similar strategy for mitigating the detrimental effects of saturation is to generate input-shaped commands that do not exceed the saturation threshold defined by $\pm p$. The

saturation mitigation condition is expressed as:

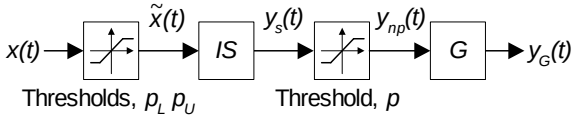
$$|y_s(t)| \leq p \tag{4.2}$$

If (4.2) is satisfied, then $y_s(t)$ is unaffected by the saturator, and the desired result that $y_{np}(t) = y_s(t)$ is achieved.

One means of satisfying the mitigation conditions for rate limiting and saturation is by conditioning the input signal, $x(t)$, in the manner depicted in the block diagrams of Figure 4.2. For the case of rate limiting, an artificial rate limiter with an upper and lower threshold range defined by m_U and m_L modifies $x(t)$ before it is filtered by the input shaper. In this way, $\widetilde{x}(t)$ actuates the input shaper instead of $x(t)$. Likewise, for the case of saturation, an artificial saturator with an upper and lower threshold defined by p_U and p_L intercepts $x(t)$ before it is filtered by the input shaper.



(a) Rate Limiting.



(b) Saturation.

Figure 4.2: Mitigating Rate Limiting and Saturation through Signal Conditioning.

In order to satisfy the mitigation conditions, sufficient values of m_U and m_L (for the case of rate limiting), and values of p_U and p_L (for the case of saturation), must be selected. However, every set of parameter values that satisfies the respective mitigation condition is not necessarily desirable. For example, suppose that p_L and p_U were each set equal to zero. While this would ensure that (4.2) was satisfied, it would also result in undesirable system performance because the system would not move. Therefore, when selecting parameter values to satisfy mitigation conditions, one must also be cognizant about how the selected parameters will impact system performance. Two methods for selecting appropriate parameter values are presented in the following sections. Each method impacts system

performance differently.

4.1.1 Method 1: Constant Parameter Selection

The artificial rate limiter places an upper and lower bound on $\dot{\tilde{x}}(t)$ such that:

$$m_L \leq \dot{\tilde{x}}(t) \leq m_U \tag{4.3}$$

The significance of this constraint on the shaped signal, $y_s(t)$, can be understood by considering the derivative of this signal:

$$\dot{y}_s(t) = A_1\dot{\tilde{x}}(t) + \cdots + A_n\dot{\tilde{x}}(t - t_n) \tag{4.4}$$

By combining (4.3) and (4.4), the following inequality can be obtained:

$$m_L \sum_{i=1}^{n} |A_i| \leq \dot{y}_s(t) \leq m_U \sum_{i=1}^{n} |A_i| \tag{4.5}$$

Values of m_U and m_L, sufficient for satisfying the mitigation condition of (4.1), can be obtained by combining (4.1) with (4.5). Solving for m_U and m_L yields:

$$m_U = -m_L = \frac{m}{\sum_{i=1}^{n} |A_i|} \tag{4.6}$$

which, for positive impulse input shapers, like the ZV and ZVD filters, simplifies to:

$$m_U = -m_L = m \tag{4.7}$$

Equation (4.6) is the *fixed rate limit constraint*. This constraint assigns constant values to the parameters, m_U and m_L, to satisfy the mitigation condition of (4.1), thereby eliminating oscillation caused by rate limiting.

The derivation of the *fixed saturation constraint* follows the same procedure as that used for obtaining the fixed rate limit constraint. This procedure yields values of p_U and p_L that are sufficient for satisfying the saturation mitigation condition of (4.2), thereby eliminating oscillation caused by saturation. The fixed saturation constraint is:

$$p_U = -p_L = \frac{p}{\sum_{i=1}^{n} |A_i|} \tag{4.8}$$

which, for positive impulse input shapers simplifies to:

$$p_U = -p_L = p \tag{4.9}$$

The fixed rate limit and fixed saturation constraints ensure that oscillation caused by rate limiting or saturation is prevented. However, these constraints are functions of m and p, the parameters associated with the actual rate limit and saturator elements, respectively. Consequently, the values of m and p may not be accurately known. Nevertheless, it may be easily shown that only conservative estimates of these parameter values are necessary for the mitigation technique to be 100% effective.

While the fixed rate limit and fixed saturation constraints are effective for eliminating oscillation caused by the respective nonlinearities, the constant thresholds resulting from the technique unnecessarily restrict some input signals. As a result, enforcing fixed mitigation constraints can cause a system to behave more slowly than necessary. The following section derives less conservative mitigation constraints by permitting the parameters of the artificial elements to be variable.

4.1.2 Method 2: Variable Parameter Selection

The rate limit mitigation condition can be expressed in terms of the derivative of the unshaped signal, $\widetilde{x}(t)$, by substituting (4.4) into (4.1). Two inequalities are obtained:

$$A_1\dot{\widetilde{x}}(t) + \cdots + A_n\dot{\widetilde{x}}(t-t_n) \leq m \tag{4.10}$$

$$A_1\dot{\widetilde{x}}(t) + \cdots + A_n\dot{\widetilde{x}}(t-t_n) \geq -m \tag{4.11}$$

Solving (4.10) and (4.11) simultaneously for $\dot{\widetilde{x}}(t)$ yields an upper and lower bound on the derivative of $\widetilde{x}(t)$:

$$r_1 \leq \dot{\widetilde{x}}(t) \leq r_2 \tag{4.12}$$

where,

$$r_1 = \frac{1}{A_1}\left[-m - \sum_{i=2}^{n} A_i\dot{\widetilde{x}}(t-t_i)\right]$$

$$r_2 = \frac{1}{A_1}\left[m - \sum_{i=2}^{n} A_i\dot{\widetilde{x}}(t-t_i)\right]$$

By enforcing (4.12), the mitigation condition of (4.1) is satisfied. In light of (4.3), equation (4.12) can be enforced by assigning the limits, r_1 and r_2, to the parameters, m_L and m_U, respectively.

Note that r_1 and r_2 vary with the derivative of $\widetilde{x}(t)$. Accordingly, (4.12) is called the *variable rate limit constraint*. The utility of this approach is that the artificial rate limiter imposes only those rate limits that are necessary for satisfying the rate limit mitigation condition. These limits automatically adjust based upon the command being issued to the system. In contrast, the fixed rate limit constraint imposes a constant set of rate limits that are sufficient, but unnecessary for all but the most extreme input signals.

The *variable saturation constraint* is derived in a similar manner as the variable rate limit constraint. This procedure yields:

$$s_1 \leq \widetilde{x}(t) \leq s_2 \tag{4.13}$$

where,

$$s_1 = \frac{1}{A_1}\left[-p - \sum_{i=2}^{n} A_i \widetilde{x}(t - t_i)\right]$$

$$s_2 = \frac{1}{A_1}\left[p - \sum_{i=2}^{n} A_i \widetilde{x}(t - t_i)\right]$$

The variable saturation constraint is enforced by assigning the limits, s_1 and s_2, to the parameters, p_L and p_U, respectively. Like the variable rate limit constraint, the variable saturation constraint imposes only those saturation limits that are necessary for satisfying the saturation mitigation condition. The limits automatically adjust with the system input.

While both the fixed and variable mitigation techniques will prevent oscillation caused by rate limiting or saturation, these techniques should not naively be implemented without a clear understanding of how other system attributes are affected, nor without an experimentally verified expectation of mitigation effectiveness. Consequently, Sections 4.1.3 through 4.1.5 provide additional insight into how fixed and variable mitigation affects system behavior. Section 4.1.3 presents an example of fixed and variable mitigation. Section 4.1.4 discusses implementation and stability issues. Section 4.1.5 presents experimental results that demonstrate the efficacy of fixed and variable mitigation. The information in these sections is presented in the context of rate limiting. However, because the mitigation techniques for rate limiting and saturation are very similar, many inferences about saturation can be made.

4.1.3 Example of Fixed and Variable Rate Limiting Mitigation

The advantage of variable rate limiting mitigation over fixed rate limiting mitigation, in terms of increasing system response, is readily apparent on systems using negative input shapers. The is because fixed rate mitigation results in an artificial rate limit that is equal to a fraction of the actual rate limit, as previously quantified in (4.6). For a UMZV shaper, fixed rate mitigation uses $m_L = -m_U = m/3$. This, in effect, causes a system to accelerate and decelerate three times more slowly than ordinary.

To demonstrate the utility of variable rate limiting compared with fixed rate limiting, three time response examples are illustrated in Figure 4.3. This figure presents simulated response data from a linear undamped second-order system, G. In the first example, labeled A, a pulse command is issued to the system. This command is filtered using a UMZV input shaper. A rate limiter with rate limit, m, corrupts the shaped command, which is subsequently issued to G. The corrupting effects of the rate limiter result in noticeable oscillation. The second and third examples, labeled B and C, respectively, are similar to the first, with the exception that an artificial rate limiter intercepts the pulse command, $x(t)$, before it is filtered by the input shaper. In B, the artificial rate limiting parameters, m_L and m_U, are selected using the fixed rate limiting mitigation equation. Thus, these values are held constant at $m_L = -m_U = m/3$. In C, the artificial rate limiting parameters are selected using the variable rate limiting mitigation equation. These values vary in real time according to (4.12).

By observation of Figure 4.3, two important points are reinforced. First, in the mitigated examples, B and C, the shaped commands that directly actuate the plant, G, are unmodified by the actual rate limit. As a result, the vibration suppression properties of these commands are not reduced, and the response of G is non oscillatory in each case. Second, for the fixed rate mitigation case, the commands throughout the signal flow rise and fall more slowly than for the variable rate mitigation case. This results in G responding more slowly to fixed rate limiting mitigation than variable rate limiting mitigation. The total latency between the two responses is labeled in the figure as L.

To further convey the differences between fixed and variable rate limiting, Figure 4.4

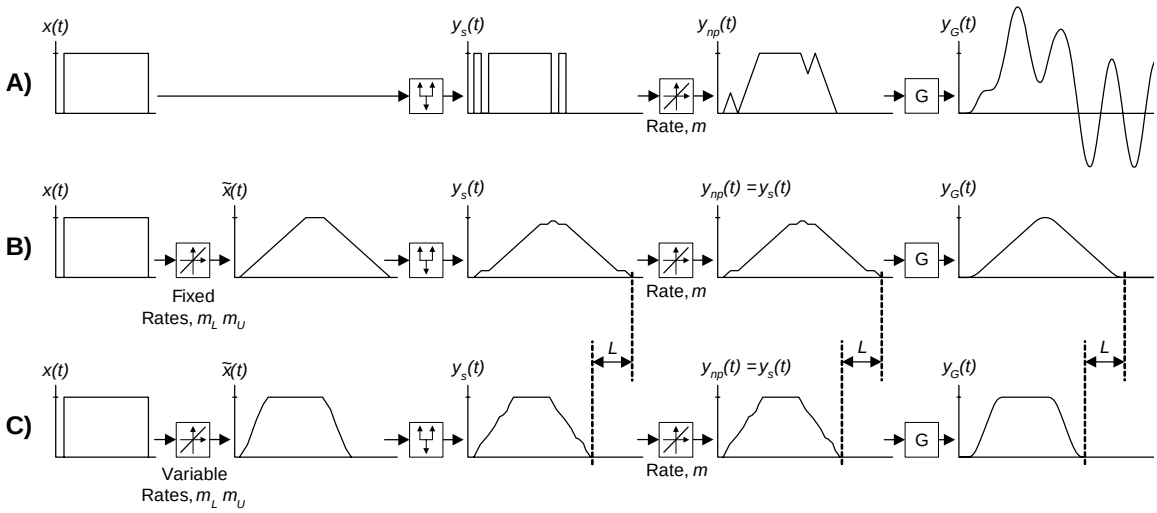


Figure 4.3: Time Response Comparison of Mitigation Techniques for Rate Limiting. A) No Mitigation Used. B) Fixed Rate Limiting Used. C) Variable Rate Limiting Used.

graphically shows the values of artificial rate limits, m_L and m_U, throughout the simulation examples, B and C. For the fixed rate case, the limits stay constant at values of $\pm m/3$. For the variable rate case, the rates change noticeably about the nominal values of $\pm \alpha m$ to ensure that the system is actuated at the maximum permissable acceleration. The significance of the coefficient, α, is discussed in the following section.

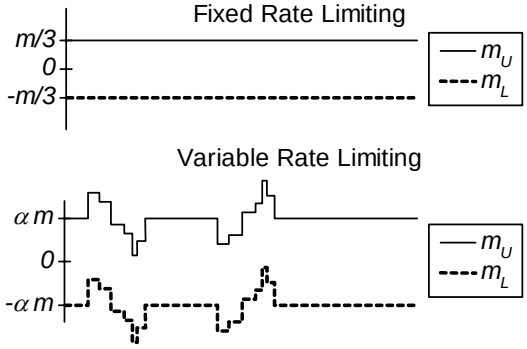


Figure 4.4: Variation of m_L and m_U for Time Response Examples, B (Upper), and C (Lower).

4.1.4 Implementation and Stability of Variable Rate Limiting

The advantage of variable rate limiting mitigation should be weighed against the difficulty of implementing the technique. In calculating variable rate limits, delayed values of the

derivative of the command signal are required. Generally, the derivative operation is implemented using numerical techniques, which can produce imprecise results and introduce latency into a signal. According to (4.12), imprecise derivative estimates can lead to imprecise values of r_1 and r_2. As a result, the mitigation condition of (4.1) may not be satisfied for all time.

Another important factor that should be considered is that the variable rate limit technique is a type of feedback control insofar as past values of a signal are used in real time to adjust current signal values. A thorough stability investigation of this dynamic aspect is beyond the scope of this work. However, this potential source of undesirable performance is noted here in the interest of disclosure.

While the potential drawbacks of imprecise derivative estimation and dynamic instability are legitimate concerns for the variable rate mitigation technique, this technique was used successfully to generate the simulated example presented here, and the experimental results presented in the subsequent section. To accommodate imprecise derivative estimation, and address potential undesirable dynamic behavior, the variable rate limit constraint of (4.12) can be implemented with a conservative scaling factor, $0.5 \lesssim \alpha \leq 1$. This coefficient can be used to conservatively calculate values of m_L and m_U:

$$m_L = \begin{cases} r_1\alpha & , \ r_1 \leq 0 \\ r_1(2-\alpha), & r_1 > 0 \end{cases} \tag{4.14}$$

$$m_U = \begin{cases} r_2\alpha & , \ r_2 \geq 0 \\ r_2(2-\alpha), & r_2 < 0 \end{cases} \tag{4.15}$$

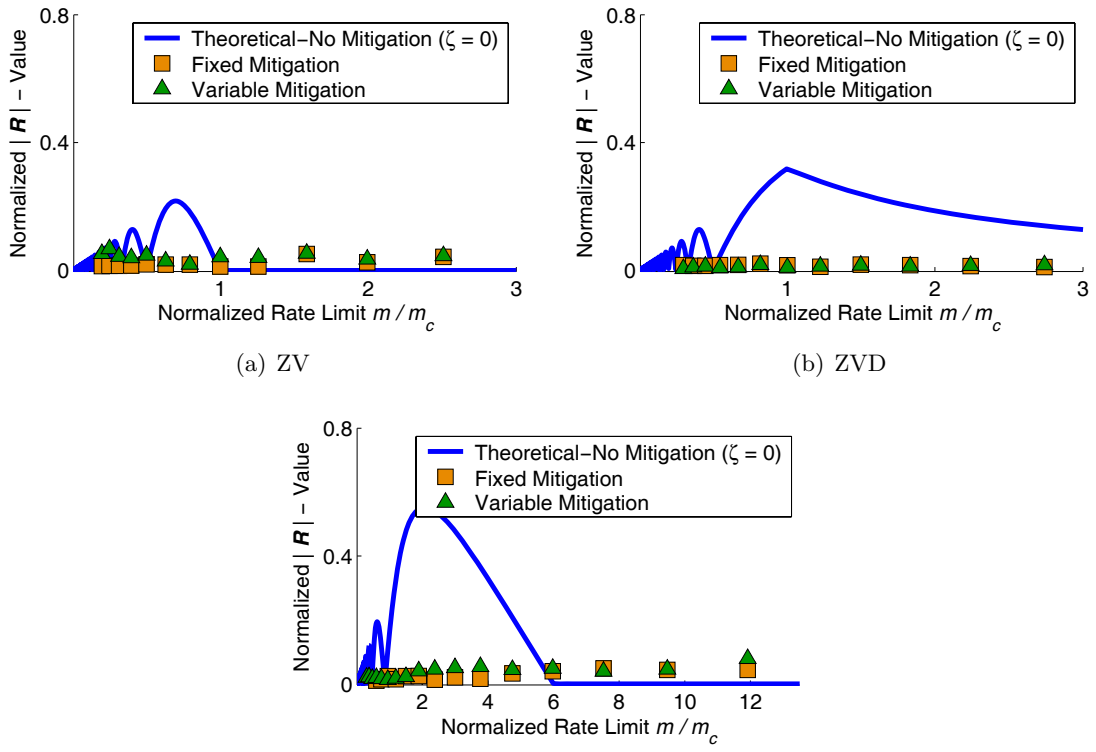
When α is equal to one, m_L and m_U are assigned values of r_1 and r_2, respectively. For lower values of α, m_L and m_U are slightly adjusted in a conservative manner to accommodate imprecise derivative estimation and potential undesirable dynamics. For the simulated results presented previously, α was set equal to 0.65. It should be noted that even with this conservative adjustment made to the variable rate limit thresholds, a more rapid system response was achieved than with the fixed rate limit technique.

4.1.5 Experimental Verification of Variable Rate Limiting

The effectiveness of the fixed and variable rate limiting mitigation techniques was experimentally evaluated using the industrial bridge crane described previously in Section 3.1.4. The procedure used to evaluate the mitigation techniques was similar to that used in Section 3.1.4 for testing the detrimental effects of rate limiting. ZV, ZVD, or UMZV shaped step commands were issued to the crane while the angular response of the payload was measured. However, for this set of experiments, Fixed or variable rate limiting mitigation was used to modify the step commands before they were filtered by the input shaper. For each trial, the rate limit, m, was set at a constant value ranging from very sluggish ($m \approx 0.25m_c$) to very rapid ($m \approx 10m_c$). The critical rate limit, m_c, was defined previously in Section 3.1.1. It is equal to the constant acceleration required to accelerate the crane from rest to maximum velocity in a time duration equal to one period of payload oscillation. The $\boldsymbol{R}$-value representing the residual oscillation from each trial was obtained by normalizing the measured results by the angular response induced by a step command.

The $\boldsymbol{R}$-values obtained from this set of experiments are presented in Figure 4.5(a) through 4.5(c) for the ZV, ZVD, and UMZV input shapers, respectively. Each figure contains three plots. The squares represent the $\boldsymbol{R}$-values obtained when fixed rate limiting mitigation was used. The triangles represent the $\boldsymbol{R}$-values when variable rate limiting mitigation was used. The solid lines represent the theoretical $\boldsymbol{R}$-values for undamped motion obtained when mitigation is not used; these curves were derived in Section 3.1, and are superimposed here for comparison purposes.

The notable feature of each graph in Figure 4.5 is that $\boldsymbol{R}$-values representing the residual oscillation when fixed or variable rate limiting mitigation was used are near zero throughout the tested range of the actual rate limit, m. This result validates the beneficial utility of the mitigation techniques at negating the detrimental effects of rate limiting on shaped commands. Furthermore, as Section 4.1.3 described, the variable rate limiting technique can do so while actuating a system more rapidly than the fixed rate limit technique.



(a) ZV

(b) ZVD

(c) UMZV

Figure 4.5: Experimentally Determined Effects of Fixed and Variable Rate Limiting Mitigation Expressed as an ***R***-Value. Theoretical ***R***-Value for Undamped Motion without Mitigation Shown with Solid Line.

4.2 Mitigating Dead-Zone and Backlash through Element Inversion

The functional relationship between the input and the output of dead-zone and backlash are well documented [118, 119, 34]. The approach used throughout the literature to minimize the problems caused by these elements on various systems utilizes their inverse functions:

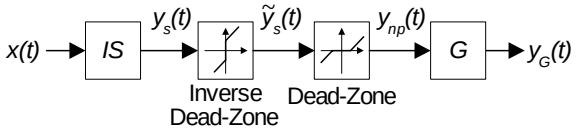
$$\text{Dead-zone: } Inverse(\beta) = \beta + \text{sign}(\beta) d_m \tag{4.16}$$

$$\text{Backlash}^2\text{: } Inverse(\beta) = \beta + \text{sign}(\dot{\beta}) b_m \tag{4.17}$$

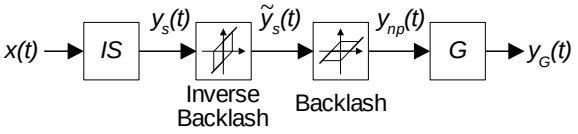
where d_m and b_m are the modeled dead-zone and backlash values of the actual dead-zone and backlash elements, respectively.

[2] A generalized inverse backlash function is described in B. Equation 4.17 is applicable only to systems in the configuration depicted previously in Figure 3.28.

To reduce the effects of these elements on the input-shaping process, inverse-backlash and inverse-dead-zone elements can be incorporated into the system in the manner depicted in Figure 4.6. The response of the inverse element to shaped commands is $\widetilde{y}_s(t)$. From the block diagrams, it is apparent that if the modeled parameters are equal to the actual parameters, then $y_{np}(t)$ is equal to $y_s(t)$ because the sequential nonlinearity and inverse-nonlinearity reduce to unity. In this scenario, the shaped signal, $y_s(t)$, acts directly on the linear plant so that residual oscillation is eliminated.



(a) Dead-Zone.



(b) Backlash.

Figure 4.6: Mitigation of Dead-Zone and Backlash through Element Inversion.

One difficulty associated with the inversion approach to mitigating dead-zone and backlash is obtaining accurate estimates of dead-zone and backlash widths for a given system. Consequently, the robustness of this mitigation technique to parameter uncertainty is considered in Section 4.2.1 for the case of dead-zone.

Backlash is not considered in the robustness analysis because, as previously discussed in Section 3.4, the functional relationship between the input and output of a backlash element and the input and output of a dead-zone element are identical, with the exception of a single attribute: backlash is active every time an input changes direction; dead-zone is active only when the input is near zero. Because of this similarity, the robustness analysis for dead-zone is also valid for backlash.

4.2.1 Robustness to Parameter Uncertainties

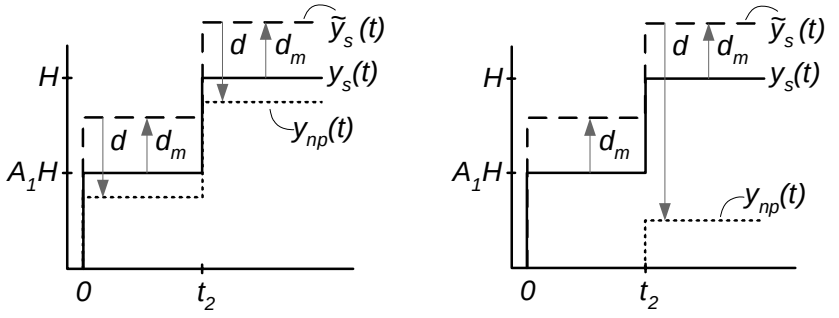
The actual dead-zone width inherent to a physical system may be difficult to accurately obtain. Therefore, it is possible that the estimated dead-zone width, d_m, can be significantly

over or under estimated. In such a scenario, the efficacy of the inversion mitigation technique can be significantly reduced, and thereby cause higher levels of oscillation than would be caused in the absence of the inversion. Consequently, the following sections address this problem with the objective of identifying upper and lower limits of parameter uncertainty that will still result in effective inversion mitigation.

4.2.1.1 ZV

To determine the robustness of the inverse mitigation approach on ZV-shaped systems, one can consider how much oscillation is induced into a given system as a result of a specified uncertainty in the dead-zone width. To this end, consider $y_{np}(t)$, the signal acting directly on G. This signal is the result of a progression of command modifications to the original ZV-shaped step command, $y_s(t)$. The first modification is caused by the inverse dead-zone block to produce $\widetilde{y}_s(t)$. This signal is further modified by the actual dead-zone to produce $y_{np}(t)$.

Figure 4.7 illustrates these modifications for different ranges of d. When d is small compared to the magnitude of the command, H, the modified commands will look similar to those shown in Figure. 4.7(a). The solid line is the original ZV-shaped step command. This command is shifted upward by the inverse-dead-zone block by a value of d_m, resulting in the command shown with the dashed line. This command is shifted downward by the dead-zone block by a value of d, producing the command shown with the dotted line. Figure 4.7(b) shows a similar modification sequence, but for larger values of d.



(a) Case 1: $0 \leq d < HA_1 + d_m$. (b) Case 2: $HA_1 + d_m \leq d < H + d_m$.

Figure 4.7: ZV-Shaped Step Command Modified by Dead-Zone and Inverse-Dead-Zone.

The oscillation caused by $y_{np}(t)$ can be determined from an $\boldsymbol{R}$-value analysis of this signal. Accordingly, the Laplace transform of the $y_{np}(t)$ signals shown in Figure 4.7 can be substituted into (2.20) to obtain the $\boldsymbol{R}$-value in terms of relative uncertainty in the dead-zone width:

$$|\boldsymbol{R}| = \begin{cases} \left|dEe^{-\zeta\omega_n t_2}\right|, & 0 \le d < HA_1 + d_m \\ |H + dE| \quad , & HA_1 + d_m \le d < H + d_m \\ 0 \qquad\quad , & H + d_m \le d \end{cases} \tag{4.18}$$

where the t_i and A_i terms are associated with the impulse times and amplitudes of the ZV shaper, and E is the relative uncertainty between d_m and d:

$$E \equiv \frac{d_m - d}{d} \tag{4.19}$$

A value of $E = 0.3$, for example, represents a 30% overestimation of the dead-zone width. A value of $E = -0.2$ represents a 20% underestimation of the dead-zone width.

The $\boldsymbol{R}$-value of $y_{np}(t)$ can alternatively be expressed in terms of the absolute uncertainty between d_m and d:

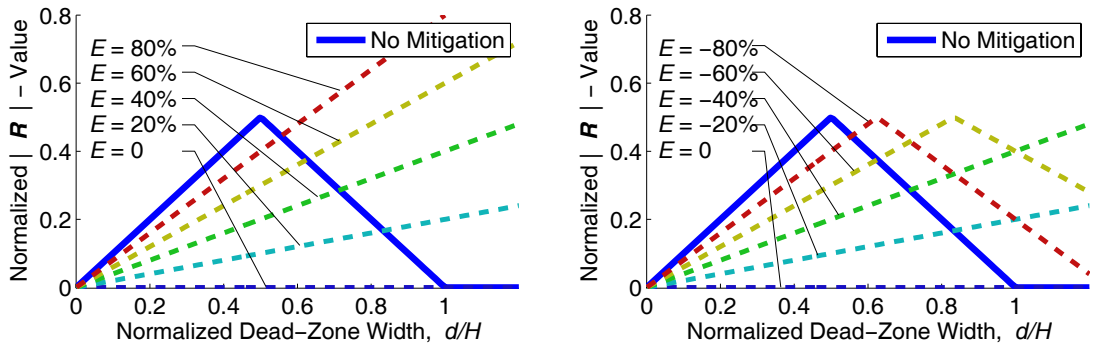
$$|\boldsymbol{R}| = \begin{cases} 0 \qquad\quad , & -\infty < D \le -H \\ |H + D| \quad , & -H < D \le -HA_1 \\ \left|De^{-\zeta\omega_n t_2}\right|, & -HA_1 < D \le d_m \end{cases} \tag{4.20}$$

where D is the absolute uncertainty between d_m and d:

$$D \equiv d_m - d \tag{4.21}$$

Equation 4.18 is plotted with the dashed lines in Figure 4.8 for various values of E and zero damping. Both the horizontal and vertical axis are normalized by H. The case when the dead-zone width is overestimated ($E > 0$) is shown in Figure 4.8(a). The case when the dead-zone width is underestimated ($E < 0$) is shown in Figure 4.8(b). Notice that the line corresponding to zero relative uncertainty indicates the anticipated result that residual oscillation will be completely eliminated for all dead-zone widths. As the relative uncertainty increases, so does the anticipated oscillatory response.

For comparison purposes, an $\boldsymbol{R}$-value plot previously shown in Figure 3.18 is also plotted in Figure 4.8. This is shown with the solid line. This line represents the oscillation caused



(a) $E > 0$. (b) $E < 0$.

Figure 4.8: Inverse-Mitigation Robustness: Relative Uncertainty, ZV-Shaper.

by dead-zone when no mitigation is used, for the case of zero damping. Thus, the graphs in Figure 4.8 concisely describes the robustness of the inverse-mitigation approach, in terms of relative uncertainty, by representing the oscillation that a given dead-zone uncertainly produces. Furthermore, Figure 4.8 also provides a means for determining when the inverse mitigation approach should and should not be used. For a given dead-zone width, the figure shows upper and lower uncertainty bounds that ensure a mitigated system reduces oscillation below levels obtained without mitigation.

A robustness graph derived in terms of absolute uncertainty is shown in Figure 4.9, where (4.20) is plotted for several different damping ratios. Along the horizontal axis, the absolute uncertainty, D, is varied. When D is less than zero, the dead-zone parameter, d, is underestimated. Likewise, when D is greater than zero, d is overestimated.

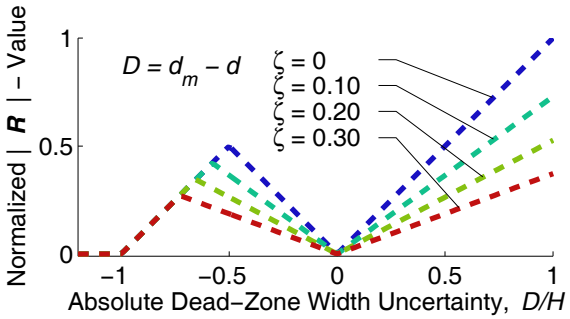


Figure 4.9: Inverse-Mitigation Robustness: Absolute Uncertainty, ZV-Shaper.

Both Figures 4.8 and 4.9 provide insight into how system oscillation varies as uncertainty in the dead-zone parameter varies.

4.2.1.2 ZVD

The robustness of the inverse mitigation technique for the case of ZVD input shaping can be determined in a similar manner as the previous analysis for the case of ZV input shaping. Oscillation induced into a given system as a result of dead-width uncertainty is determined by considering $y_{np}(t)$, the signal acting directly on G. The progression of command modifications that produce $y_{np}(t)$ are very similar to those shown previously in Figure 4.7 for the ZV input shaper.

The oscillation caused by $y_{np}(t)$ can be determined from an $\boldsymbol{R}$-value analysis of this signal. The $\boldsymbol{R}$-value is expressed here in terms of relative uncertainty, E, in the dead-zone width:

$$|\boldsymbol{R}| = \begin{cases} e^{-\zeta\omega_n t_3}\left|(A_1H + dE) + A_2He^{t_2Q} + A_3He^{t_3Q}\right|, & 0 \le d < A_1H + d_m \\ e^{-\zeta\omega_n t_2}\left|(A_1 + A_2)H + dE + A_3He^{t_2Q}\right| & , \ A_1H + d_m \le d < (A_1 + A_2)H + d_m \\ |H + dE| & , \ (A_1 + A_2)H + d_m \le d < H + d_m \\ 0 & , \ H + d_m \le d \end{cases} \tag{4.22}$$

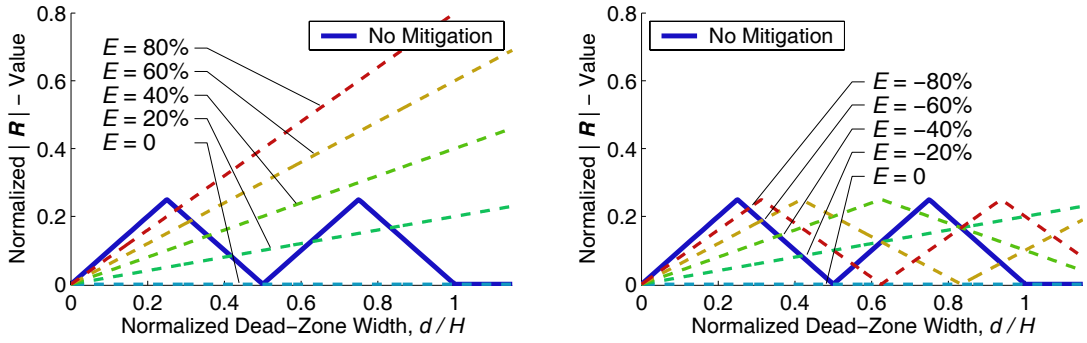
where the t_i and A_i terms are associated with the impulse times and amplitudes of the ZVD shaper, and $Q = \zeta\omega_n + j\omega_d$.

Alternatively, the $\boldsymbol{R}$-value of $y_{np}(t)$ can be expressed in terms of the absolute uncertainty, D, between d_m and d:

$$|\boldsymbol{R}| = \begin{cases} e^{-\zeta\omega_n t_3}\left|(A_1H + D) + A_2He^{t_2Q} + A_3He^{t_3Q}\right|, & -HA_1 < D \le d_m \\ e^{-\zeta\omega_n t_2}\left|(A_1 + A_2)H + D + A_3He^{t_2Q}\right| & , \ -H(A_1 + A_2) < D \le -HA_1 \\ |H + D| & , \ -H < D \le -H(A_1 + A_2) \\ 0 & , \ D \le -H \end{cases} \tag{4.23}$$

Equation 4.22 is plotted with the dashed lines in Figure 4.10 for various values of E and zero damping. Again, the horizontal and vertical axis are normalized by H. The case when the dead-zone width is overestimated is shown in Figure 4.10(a). The case when the dead-zone width is underestimated is shown in Figure 4.10(b).

In order to provide a means for determining when the inverse mitigation approach should



(a) $E > 0$. (b) $E < 0$.

Figure 4.10: Inverse-Mitigation Robustness: Relative Uncertainty, ZVD-Shaper.

and should not be used, an $\boldsymbol{R}$-value plot previously shown in Figure 3.19 is also plotted in Figure 4.10. This is shown with the solid line. This line represents the oscillation caused by dead-zone when no mitigation is used, for the case of zero damping. For a given dead-zone width, the graphs in Figure 4.10 show upper and lower uncertainty bounds that ensure a mitigated system reduces oscillation below levels obtained without mitigation.

Figure 4.11 provides insight into the robustness of the inverse mitigation approach in terms of absolute uncertainty. In this figure, (4.23) is plotted for several different damping ratios as the absolute uncertainty, D, is varied.

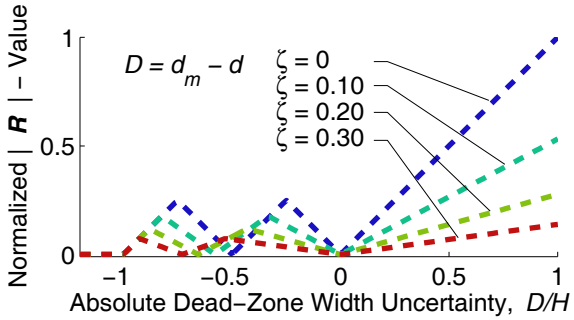


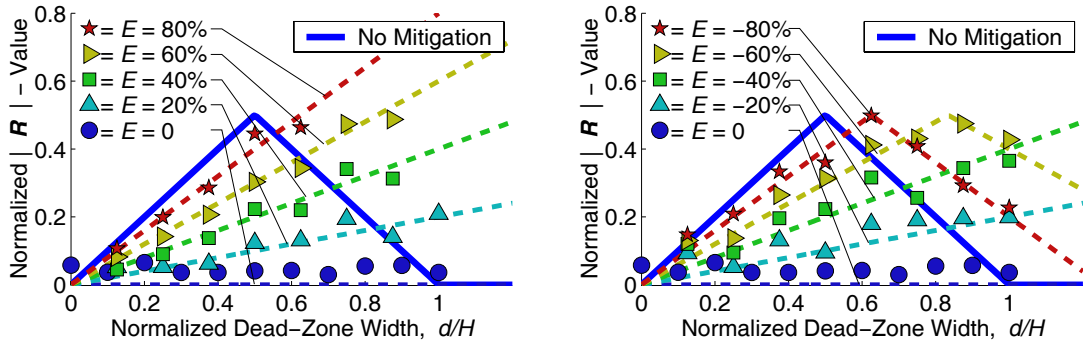
Figure 4.11: Inverse-Mitigation Robustness: Absolute Uncertainty, ZVD-Shaper.

4.2.2 Experimental Verification of Mitigation and Robustness

The results of the robustness analysis were verified by using the crane described previously in Section 3.1.4. Several known dead-zone widths, d, were manually programmed into the crane drives ranging from $d = 0$ to $d = H$. For each dead-zone width, a series of ZV and

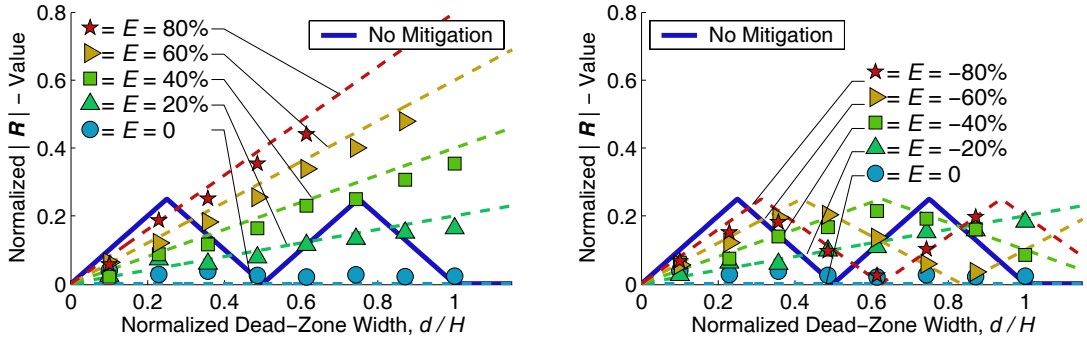
ZVD-shaped step commands were issued to the crane. These commands were modified by an inverse-dead-zone function with estimated dead-zone width, d_m. The residual oscillation caused by each command was measured, then normalized by the residual oscillation amplitude caused by a velocity step command of magnitude H.

To show the measured results from this series of trials, the graphs of Figures 4.8 and 4.10 are replotted in Figures 4.12 and 4.13, respectively, with the addition of the experimental data. The similarity between the measured and predicted results validates the preceding robustness analysis.



(a) $E > 0$. (b) $E < 0$.

Figure 4.12: Experimental and Theoretical Inverse-Mitigation Robustness, ZV-Shaper.



(a) $E > 0$. (b) $E < 0$.

Figure 4.13: Experimental and Theoretical Inverse-Mitigation Robustness, ZVD-Shaper.

4.3 Conclusion

The detrimental effects of rate limiting, saturation, dead-zone, and backlash on input-shaped systems are manifest in residual system oscillation. The strategy for mitigating oscillation

caused by saturation and rate limiting involves conditioning input commands with an artificial saturator or rate limiter. Fixed parameter constraints assign constant parameter values to artificial elements. Variable parameter constraints permit the parameter values of the artificial elements to vary according to the command being issued to the system. Although more difficult to implement, variable parameter constraints offers advantages over fixed parameter constraints by increasing the rapidity at which a system may respond to issued commands. To mitigate the oscillation caused by dead-zone and backlash, an inversion technique was proposed. The robustness of the approach to uncertainties in dead-zone and backlash width was analyzed. Experiments on an industrial bridge crane were used to validate key results, namely, 1) signal conditioning is an effective way to reduce oscillation caused by saturation or rate limiting, and 2) inverse-mitigation of dead-zone and backlash can be effective even with large parameter uncertainty.

CHAPTER V

FINITE-STATE INPUT SHAPING[1]

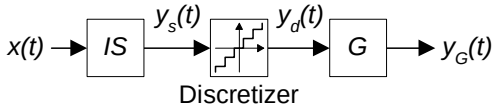
CHAPTER SUMMARY: *Input shaping works well on systems with continuously-variable actuation because the system is capable of tracking a large variety shaped commands. However, many systems only have a finite number of actuation states. For example, some relay driven cranes have three actuation states: off, half-speed, and full speed. The discretizing effect of finite-state actuation can reduce the effectiveness of input shaping. This chapter investigates the detrimental effects of discretization. The operational effects of finite-state input shaping are evaluated by using experimental results from a human-operated bridge crane, and a 3-dimensional simulation of this crane.*

Finite-state actuation describes a class of actuator nonlinearity that arises when systems can only be propelled at discrete levels of actuator effort. Thruster-jet controlled satellites, for example, have two actuation states per thruster jet: on and off. Similarly, some cranes employ relay-driven drives that actuate the system with three states: off, low-speed, and full-speed. Other examples of finite-state actuation are found in digital control applications where low-bit analog-to-digital conversion results in coarsely discretized signals [94].

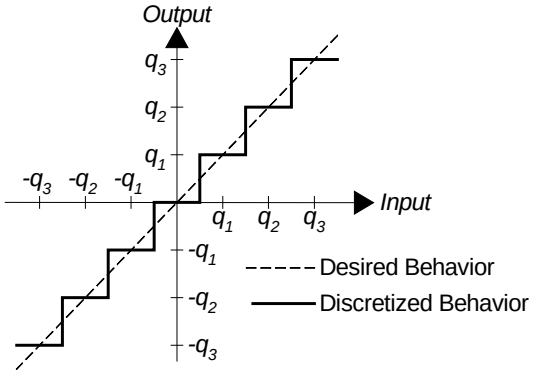
The discrete behavior of finite-state actuators can significantly reduce the capability of input-shaping filters to eliminate oscillation. This is because the finite-state actuators cannot track the shaped commands. Figure 5.1(a) illustrates the conventional implementation of input shaping on a system with finite-state actuation. The behavior of the actuator is modeled with a discretization element, which corrupts the shaped signals before they actuate the linear plant. Figure 5.1(b) illustrates the functional relationship between the input and output of a discretization element. The dashed line represents the desired linear

[1]Portions of this chapter are published in [106].

relationship, and the the staircase function is the actual behavior. The discrete levels of the staircase function correspond to the quantization levels of the discretizer.



(a) Serial Input Shaper, Discretizer, and Linear Plant.



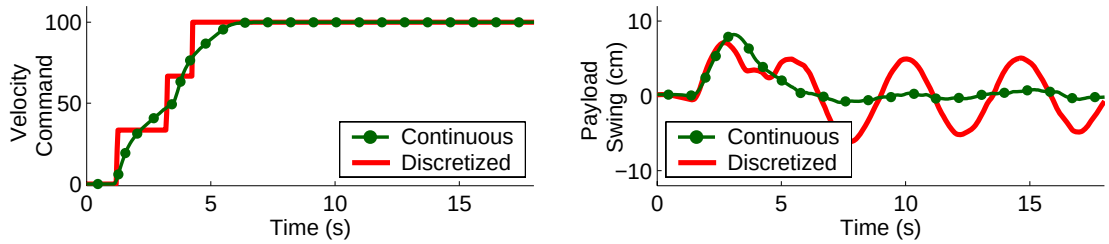
(b) Functional Relationship Between the Input and Output of a Discretizer.

Figure 5.1: Input Shaping Implemented on a System with a Discretization Element.

To illustrate the adverse effects of finite-state actuation on input-shaping control, two experiments were conducted on the 10-ton bridge crane described previously in this thesis. Recall that the crane is equipped with vector drives and induction motors that are capable of tracking continuously-variable velocity commands.

In the first experiment, the continuously-variable velocity command shown with the dotted line in Figure 5.2(a) was issued to the crane. The command was generated by convolving a two-impulse input shaper with a reference command that would accelerate the system to full velocity. The input shaper was designed to eliminate residual payload swing. The measured response of the payload to the issued command is shown with the dotted line in Figure 5.2(b). As anticipated, very little payload swing was exhibited after the initial deflection.

In the second experiment, a discretized version of the input-shaped velocity command was issued to the crane. The discretization levels were at 0-velocity, 1/3-velocity, 2/3-velocity, and full-velocity. The purpose of issuing the discretized version of the command



(a) Continuous and Discretized Velocity Commands. (b) Payload Swing in Response to Velocity Commands.

Figure 5.2: Crane Behavior for Continuous and Finite-State Actuation.

was to simulate the behavior of a crane that uses relay-driven drives with the corresponding actuation states. The discretized command and measured payload-swing response are shown with the solid lines in Figures 5.2(a) and 5.2(b), respectively. It is evident that the coarsely discretized command resulted in substantially more oscillation than the original shaped command.

Very few researchers have addressed the adverse effects of finite-state actuation on input shaping. Peláez and Singhose considered the problem of implementing input shaping on low-cost digital controllers [67]. However, their work focused on finely discretized signals that were nearly identical to the continuous counterparts. Kavanagh addressed the problem in the context of FIR filters. He showed that when an FIR filter is used as a differentiator, the signal quantization can lead to poor derivative estimates [38].

Because a thorough investigation of finite-state input shaping is lacking, the objective of this chapter is to expand the body of knowledge pertaining to this subject. Special attention is given to finite-state input shaping in the context of the ZV, ZVD, and UMZV input shapers. Specifically, this chapter seeks to:

1. *Establish a benchmark finite-state system.*

2. *Identify important properties of the benchmark finite-state system that will provide a framework for further analyses.*

3. *Evaluate the operational effects of discretizing ZV, ZVD, and UMZV-shaped signals.*

The following section addresses objective number one and two. The third objective is addressed in Section 5.2.

5.1 Input Shaping with Finite States

Consider the system schematically illustrated in Figure 5.3. In this block diagram, G represents a linear plant, which is to be controlled. D represents the nonlinear functionality of the system drives and motors. For finitely-actuated systems, this block discretizes the control signals, $x(t)$, so that the actuating signal, $y_d(t)$, can only assume the admissible quantization levels of D. Or, more concisely:

$$y_d(t) \in \mathbf{Q} \tag{5.1}$$

where $\mathbf{Q}$ is the set of m admissible actuation states:

$$\mathbf{Q} = \{q_1, q_2, \ldots, q_m\} \tag{5.2}$$

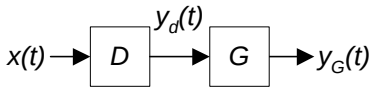


Figure 5.3: Finite-State Actuation of a Linear System.

One class of finitely-actuated systems use control signals that already assume admissible values of $\mathbf{Q}$, obviating the function of the discretizer in D. Pendent controlled cranes, for example, issue control signals from a pendent in discrete directions and velocities that correspond to the various pendent button states. If the pendent button states are equal to the actuation states, then:

$$x(t) \in \mathbf{Q} \tag{5.3}$$

Input shaping is implemented on a system of this type in the manner depicted in the block diagram of Figure 5.4. This class of input-shaped systems is the primary focus of this chapter. Notice that both the control signal, $x(t)$, and actuation signal, $y_d(t)$, are comprised of admissible states, $\mathbf{Q}$. However, the shaped control signal, $y_s(t)$, is comprised of states $\mathbf{Q}_s$, which may not be the same set as $\mathbf{Q}$.

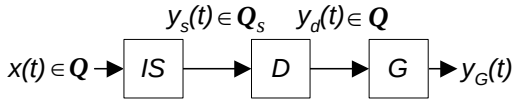


Figure 5.4: Finite-State Input Shaping.

The convolution relationship that produces the shaped signal, $y_s(t)$, was previously derived in Chapter 1. It is reproduced here for convenience:

$$y_s(t) = A_1 x(t) + A_2 x(t - t_2) + \cdots + A_n x(t - t_n), \quad x(t < 0) = 0 \tag{5.4}$$

By examining this equation, it is evident that $y_s(t)$ assumes values that depend on both $x(t)$, and the impulse amplitudes, A_i. This set of input-shaped states is defined as:

$$\mathbf{Q}_s = \{q_{s1}, q_{s2} \ldots\} \tag{5.5}$$

If the elements of $\mathbf{Q}_s$ are different from the elements of $\mathbf{Q}$, then the input-shaped commands can assume values that are not achievable by the system actuators. In this scenario, the shaped commands will be modified by the discretizer in D to conform to the admissible states. As a result, oscillation may not be fully suppressed.

As a first step toward a more thorough understanding of this situation, it is useful to identify the elements of $\mathbf{Q}_s$. This is accomplished in a general sense in Section 5.1.1. The special case were zero is an element in $\mathbf{Q}$ is also considered. In Section 5.1.2, an example is presented where the elements of $\mathbf{Q}_s$ are calculated for the ZV, ZVD, and UMZV input shapers and a given set of admissible states. By close examination of the resulting $\mathbf{Q}_s$ sets, some generalizations are made about the performance of the different input shapers on finitely-actuated systems.

5.1.1 Identification of Input-Shaped States

The elements of the set $\mathbf{Q}_s$ can be determined from (5.4). An expanded form of this equation is:

$$y_s(t) = \begin{cases} A_1 x(t), & 0 \le t < t_2 \\ A_1 x(t) + A_2 x(t - t_2), & t_2 \le t < t_3 \\ \quad \vdots & \quad \vdots \\ A_1 x(t) + A_2 x(t - t_2) + \cdots + A_n x(t - t_n), & t_n \le t \end{cases} \tag{5.6}$$

To facilitate an analysis of this equation, the following set and set operator are defined:

Set: $\mathbf{S}_i$

$$\mathbf{S}_i \equiv A_i\mathbf{Q} \tag{5.7}$$

Set Operator: $\hat{\times}$

$$\mathbf{X}\hat{\times}\mathbf{Y} \equiv \{(x+y)|\, x \in \mathbf{X} \text{ and } y \in \mathbf{Y}\} \tag{5.8}$$

The $\hat{\times}$ operator is a modified Cartesian product that combines two sets, $\mathbf{X}$ and $\mathbf{Y}$, to produce a third set. The elements in the resulting set are comprised of the sum of all possible ordered pairs whose first component is a member of $\mathbf{X}$, and whose second component is a member of $\mathbf{Y}$. For example:

$$\{0,1,2\}\hat{\times}\{a,b\} = \{0+a, 0+b, 1+a, 1+b, 2+a, 2+b\}$$

The modified Cartesian product operation can be visualized using a nodal network like the one shown in Figure 5.5. Each path originating from the first column of nodes and terminating at the last column of nodes represents an element in the resulting set.

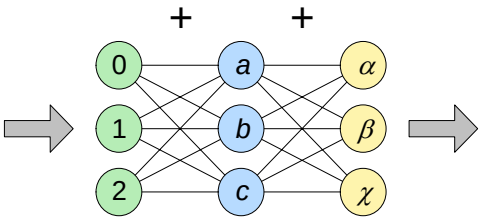


Figure 5.5: Nodal Network of the Modified Cartesian Product: $\{0,1,2\}\hat{\times}\{a,b,c\}\hat{\times}\{\alpha,\beta,\chi\}$.

From (5.6), it is evident that during the time interval between $t = 0$ and $t = t_2$, the possible values of $y_s(t)$ are $A_1x(t)$. But, $x(t) \in \mathbf{Q}$. Therefore, $\mathbf{S}_1 = A_1\mathbf{Q}$ must be a subset of $\mathbf{Q}_s$:

$$A_1x(t) \in \mathbf{S}_1 \subset \mathbf{Q}_s \tag{5.9}$$

Similarly, during the second time interval between $t = t_2$ and $t = t_3$, the possible values of $y_s(t)$ are obtained by adding elements of $\mathbf{S}_1$ to elements of $\mathbf{S}_2 = A_2\mathbf{Q}$. The resulting

set is also a subset of $\mathbf{Q}_s$, and can be expressed succinctly by using the modified Cartesian product operator:

$$A_1 x(t) + A_2 x(t - t_2) \in \left\{ \mathbf{S}_1 \hat{\times} \mathbf{S}_2 \right\} \subset \mathbf{Q}_s \tag{5.10}$$

Successive analyses of the remaining time intervals in (5.6) reveal that $\mathbf{Q}_s$ is comprised of n subsets:

$$\mathbf{Q}_s = \left\{ \{\mathbf{S}_1\}, \{\mathbf{S}_1 \hat{\times} \mathbf{S}_2\}, \ldots, \{\mathbf{S}_1 \hat{\times} \mathbf{S}_2 \hat{\times} \cdots \hat{\times} \mathbf{S}_n\} \right\} \tag{5.11}$$

or more succinctly,

$$\mathbf{Q}_s = \{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_n\} \tag{5.12}$$

where,

$$\mathbf{P}_k = \left\{ \mathbf{S}_1 \hat{\times} \mathbf{S}_2 \hat{\times} \cdots \hat{\times} \mathbf{S}_k \right\} \tag{5.13}$$

Because each set, $\mathbf{S}_i$, contains m elements, the total number of elements contained in $\mathbf{P}_k$ is obtained from the fundamental theory of counting to yield:

$$\text{Size}(\mathbf{P}_k) = m^k \tag{5.14}$$

Therefore, the number of elements contained in $\mathbf{Q}_s$ may be found from the geometric series:

$$\text{Size}(\mathbf{Q}_s) = m + m^2 + \cdots + m^n = \frac{m - m^{(n+1)}}{1 - m} \tag{5.15}$$

The value of each element defined in (5.11) represents a possible value of the input-shaped signal, $y_s(t)$. Furthermore, in light of the amplitude constraint (1.4), which enforces that the sum of the impulse amplitudes equals unity, it may shown that the elements of $\mathbf{Q}$ are contained in the elements of $\mathbf{Q}_s$:

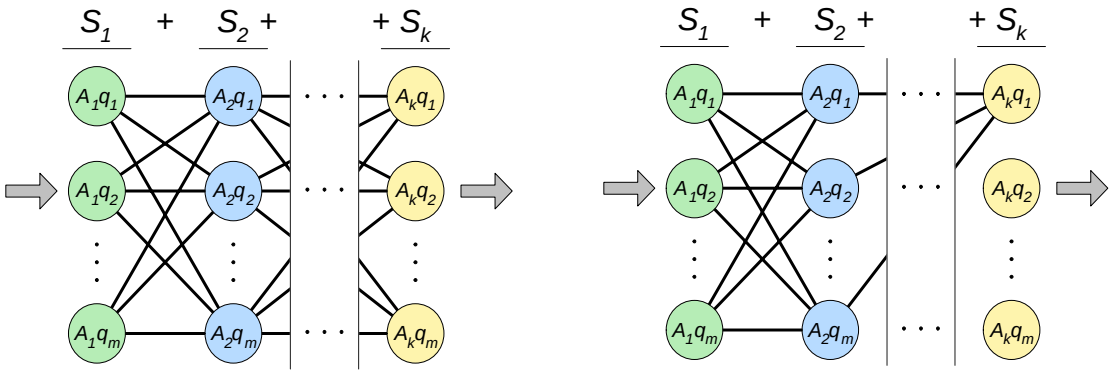
$$\mathbf{Q} \subset \mathbf{Q}_s \tag{5.16}$$

5.1.1.1 Special Case: $0 \in \mathbf{Q}$

The special case when zero is an admissible state corresponds to zero being contained in the set $\mathbf{Q}$. This occurrence is significant. It indicates quiescence (non activity) in the actuation of a system. For most physical systems "zero" actuation is always admissible, therefore, this case deserves special attention.

Consider the set $\mathbf{P}_k$, graphically represented by the nodal network in Figure 5.6(a). Each element in $\mathbf{P}_k$ has k terms: one term from each of the k columns $\mathbf{S}_1$ through $\mathbf{S}_k$. Now consider the set of elements within $\mathbf{P}_k$ that end with the term $A_k q_1$. This set is graphically represented by the nodal network in Figure 5.6(b) and denoted by $\mathbf{P}_{kq_1}$. It is equal to the modified Cartesian product of $\mathbf{P}_{k-1}$ and $A_k q_1$:

$$\mathbf{P}_{kq_1} = \mathbf{P}_{k-1} \hat{\times} A_k q_1 \tag{5.17}$$



(a) Nodal Network Representing $\mathbf{P}_k$.

(b) Nodal Network Representing $\mathbf{P}_{kq_1}$.

Figure 5.6: Nodal Networks Representing Modified Cartesian Products.

Because every element in $\mathbf{P}_{kq_1}$ is also in $\mathbf{P}_k$, it is apparent that:

$$\mathbf{P}_{kq_1} \subset \mathbf{P}_k \tag{5.18}$$

Now, without loss of generality, suppose that:

$$q_1 \equiv 0 \tag{5.19}$$

Then, it follows from (5.17) that:

$$\mathbf{P}_{kq_1} = \mathbf{P}_{k-1} \hat{\times} 0 = \mathbf{P}_{k-1} \tag{5.20}$$

Thus,

$$\mathbf{P}_{k-1} \subset \mathbf{P}_k \tag{5.21}$$

which implies that:

$$\mathbf{P}_1 \subset \mathbf{P}_2 \subset \cdots \subset \mathbf{P}_n \tag{5.22}$$

Therefore, given the expression for $\mathbf{Q}_s$ in (5.12), it is evident that if zero is an element of $\mathbf{Q}$, then $\mathbf{P}_1$ through $\mathbf{P}_{n-1}$ contain redundant elements. As a result, $\mathbf{Q}_s$ may be redefined as:

$$\mathbf{Q}_s = \mathbf{P}_n \tag{5.23}$$

This result is summarized as:

Theorem 1

Let $\mathbf{Q} = \{q_1, q_2, \ldots, q_m\} \subset \mathbb{R}^1$, *where* $0 \in \mathbf{Q}$.

Let $\mathbf{A} = \{A_1, A_2, \ldots, A_n\} \subset \mathbb{R}^1$.

Define $\mathbf{P}_k = A_1\mathbf{Q}\hat{\times}A_2\mathbf{Q}\hat{\times}\cdots\hat{\times}A_k\mathbf{Q}$.

Then $\mathbf{P}_1 \subset \mathbf{P}_2 \subset \cdots \subset \mathbf{P}_n$.

5.1.2 Admissible and Non-Admissible State Distribution

This section considers an example finite-actuation system where the admissible actuation states, $\mathbf{Q}$, are evenly distributed and symmetric about zero. This class of actuation states is significant because it represents a large class of finitely-actuated systems.

The objective of this example is to calculate the input-shaped states, $\mathbf{Q}_s$, for the ZV, ZVD, and UMZV input shapers, and to consider how the elements of $\mathbf{Q}$ are distributed among the elements of $\mathbf{Q}_s$. These results will provide insight into how the ZV, ZVD, and UMZV input shapers are affected by finite-state actuation.

Suppose that a ZVD input shaper is designed for a lightly damped system with $\zeta \approx 0$. Then, according to (1.10), $A_1 = A_3 = 1/4$, and $A_2 = 1/2$. Also, suppose that $\mathbf{Q}$ contains nine elements that are evenly distributed and symmetric about zero such that:

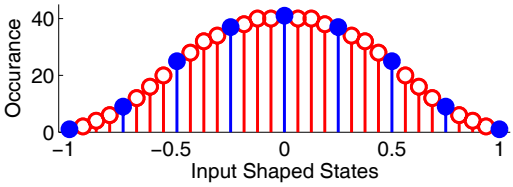
$$\mathbf{Q} = \left\{0,\ \pm\frac{1}{4},\ \pm\frac{1}{2},\ \pm\frac{3}{4},\ \pm 1\right\} \tag{5.24}$$

Then, from (5.24) and (5.7), the elements of sets $\mathbf{S}_1$ through $\mathbf{S}_3$ can be obtained:

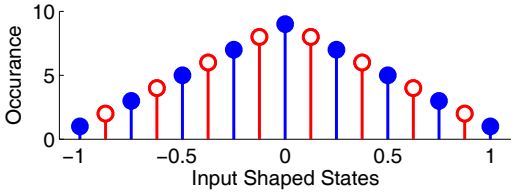
$$\mathbf{S}_1 = \mathbf{S}_3 = \left\{0,\ \pm\frac{1}{16},\ \pm\frac{1}{8},\ \pm\frac{3}{16},\ \pm\frac{1}{4}\right\} \tag{5.25}$$

$$\mathbf{S}_2 = \left\{0,\ \pm\frac{1}{8},\ \pm\frac{1}{4},\ \pm\frac{3}{8},\ \pm\frac{1}{2}\right\} \tag{5.26}$$

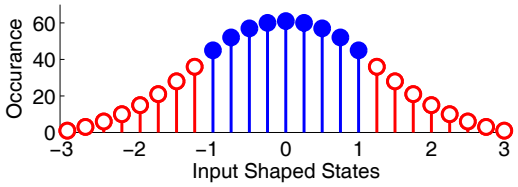
Because zero is an element of $\mathbf{Q}$, (5.23) can be used to determine the elements of $\mathbf{Q}_s$. According to (5.14), there are $9^3 = 729$ elements in $\mathbf{Q}_s$. However, the values of these elements are not unique. This is clearly seen in Figure 5.7(a). This figure displays a histogram of the elements of $\mathbf{Q}_s$. The horizontal axis of this figure indicates the values of the elements of $\mathbf{Q}_s$. The vertical axis indicates the number of elements in $\mathbf{Q}_s$ that equal the corresponding value. The solid circles on the plot denote admissible actuation states. Hollow circles denote non-admissible states.



(a) ZVD Shaping of $\mathbf{Q}$ in (5.24).



(b) ZV Shaping of $\mathbf{Q}$ in (5.24).



(c) UMZV Shaping of $\mathbf{Q}$ in (5.24).

Figure 5.7: Histogram of the Elements Contained in $\mathbf{Q}_s$. Admissible States (Solid Circles). Non-Admissible (Hollow Circles)

Identical procedures have been conducted to find the $\mathbf{Q}_s$-distributions for the ZV, and UMZV input shapers. These results are shown in Figures 5.7(b) and 5.7(c), respectively. Similar, but more dense, distributions would have been produced if, for this example, the number of admissible states in $\mathbf{Q}$ was larger.

There are several significant features in these distributions. First, it is evident in each of the distributions that all the values of $\mathbf{Q}$ are contained in $\mathbf{Q}_s$. This is because, as noted previously, $\mathbf{Q} \subset \mathbf{Q}_s$.

Second, the distribution for the UMZV shaper is noticeably different than the distributions for either the ZV or ZVD shapers. In the case of the UMZV distribution, all the admissible states are grouped together in the center of the curve, whereas for the positive impulse shapers, the admissible states are evenly distributed throughout the curve.

Finally, the histogram plots resemble bell-shaped or triangular-shaped curves. In either case, this indicates that the modified Cartesian product of the sets $\mathbf{S}_i$ produce more elements with values near the center of the distributions than values near the extrema of the distributions. This is significant because of what it implies about issued commands: Generally speaking, reference commands issued to a ZV, ZVD, or UMZV shaper will tend to produce more $\mathbf{Q}_s$ values near the distribution center than near the distribution extrema. In light of how the admissible states are grouped among the non-admissible states, this fact suggests that the UMZV input shaper is less affected by finite-state actuation than either the ZV or ZVD input shapers. This assertion is made because, for the UMZV case, reference commands tend to produce admissible states more frequently than non-admissible states. This cannot be said, however, for either of the positive-impulse shapers. Experiments presented in the following section support this conclusion.

5.2 Oscillatory Effects of Finite-State Actuation

It has been shown that finite-state actuation can produce a significant number of non-admissible actuation states, and that the distribution of the non-admissible states among the admissible states is different for different input shapers. Yet, several questions pertaining to the operational effects of finite-state actuation remain:

1. *Does the oscillation caused by finite-state actuation detrimentally affect system performance?*

2. *How is the amplitude of system oscillation related to the number of actuation states?*

3. *Is the UMZV input shaper really more resistant to the detrimental effects of finite-state actuation than the ZV or ZVD input shapers?*

The answers to these questions depend largely upon the behavior of the reference commands being issued to a system, and are therefore application dependent. Nevertheless, this section addresses these questions for one specific control application: human operation of a bridge crane with ZV, ZVD, and UMZV input shaping. Given the results for this application, inferences about the performance of similar systems may be made.

5.2.1 Description of the Crane and Crane Simulation

Experiments were conducted on the 10-ton bridge crane described previously in Section 3.1.4. A photograph of this crane is shown in Figure 5.8(a). Similar experiments were also conducted by using a 3-dimensional computer simulation of the 10-ton crane. A screen-shot of this simulation is shown in Figure 5.8(b).

(a) 10-Ton Bridge Crane. (b) Screen-Shot of Crane Simulation.

Figure 5.8: Experimental Test Beds for Crane Operator Experiments.

While the 10-ton crane is a valuable resource for gathering realistic industrial data, there are also some drawbacks to this testing venue. First, it can be time consuming to program the crane with different experimental control algorithms. Secondly, volunteer crane operators must be recruited, safety-trained, and then scheduled to participate. This logistical bottleneck means that obtaining statistically significant data sets can take several weeks, and hundreds of hours. Therefore, the experiments conducted during this study were completed by operators using both the industrial crane and the crane simulator. By using the actual and simulated cranes as test beds, large data sets were obtained easily. While 17 experimental trials from 3 volunteers were gathered by using the actual crane, over 100

trials from 37 volunteers were gathered by using the simulation.

The 10-ton crane is equipped with programmable vector drives and induction motors, a programmable logic controller (PLC), and a joystick interface. This system was configured to behave like the benchmark finite-actuation system previously shown in Figure 5.4. The behavior of relay-driven crane motors was mimicked by programming the drives to respond with a finite number of actuation states. The joystick interface was configured to issue only admissible commands to the PLC. The input-shaping algorithm was implemented on the PLC, from where the shaped commands were then issued to the drives and motors.

Controlling the simulated crane is similar to controlling the industrial crane. A joystick-equipped computer allows the simulation to be manipulated in various directions and velocities that correspond to the admissible actuation states.

5.2.2 Description of the Experiments

For each experimental trial, the crane was enabled with a ZV, ZVD, or UMZV input shaper. Also, a finite number of admissible actuation states was programmed into the drives. The actuation states were evenly distributed and symmetric about zero. The fewest number of states was 3 (-full velocity, stop, full velocity). The largest number of states was 15. Additionally, trials were conducted while the crane was permitted to run in continuously-variable velocity mode. This is equivalent to an infinite number of actuation states between the permissable range of velocities.

Each experimental trial consisted of an operator driving the crane toward a target region. Every twenty seconds, the location of the target region changed to a different location. The objective of the operator was to continually position the crane so that the payload remained within the moving target region for as long as possible. A complete trial consisted of 15 different target locations.

The skill level operators participating in the studies varied from novice to intermediate. Participants using the actual crane were required to undergo a short safety and crane operation training that included driving the crane to several locations. During the training, the crane was in manual mode, and did not have input shaping enabled. Participants using

the crane simulator did not undergo any special training.

During each trial conducted on the 10-ton crane, the payload suspension cable remained fixed at a nominal operating length of 6 m, giving the system an oscillation period of approximately 5 s. The damping ratio was approximately 0.01. Each trial conducted on the crane simulator used a payload suspension cable length of 4 m and a damping ratio of 0.01, giving the simulated system a period of approximately 4 s.

Approximately one-third of the trials utilized the ZV input shaper, one-third of the trials utilized the ZVD input shaper, and the final third utilized the UMZV input shaper.

Several state variables were recorded during each experiment. These included the angular displacement of the payload, the position and velocity of the trolley, and the commanded velocity signals. These quantities were readily available from the simulation. But, for the industrial crane, a machine vision system, in conjunction with the PLC, was used to record these states.

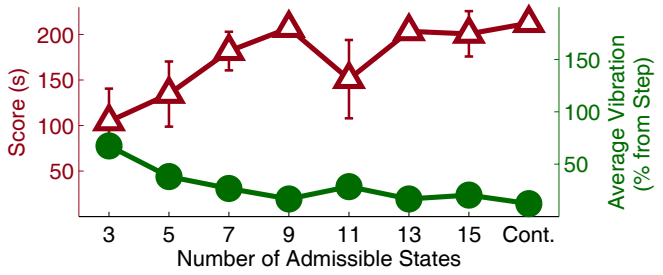
5.2.3 Experimental Results

In order to assess the operational effects of finite-state actuation on input shaping, two metrics were used to quantify system performance. The first was the time, in seconds, that the payload was contained within the target region. Higher values of this *score* indicate efficient crane operation, both in terms of gross motion, and reduced payload swing. The second metric used to gauge system behavior was the average angular displacement of the payload over the course of the trial[2]. Although there are several elements that contribute to total system oscillation, such as external disturbances and system nonlinearities, this metric still provides an indication of how much oscillation is caused when the input-shaping filter produces non-admissible states.

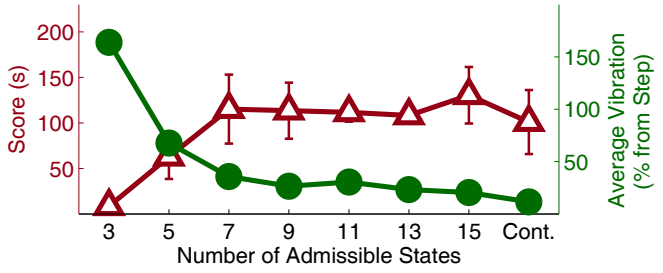
The targeting score and vibration results obtained from the crane simulator are summarized in the plots of Figure 5.9. The triangular markers indicate the average scores obtained by all the operators for each different testing configuration. The circular markers

[2]It should be noted that this measurement differs fundamentally from average *residual* angular displacement. Residual oscillation refers to payload sway exhibited after a command has reached a steady-state value. Total oscillation includes payload sway during transient and steady-state portions of a command.

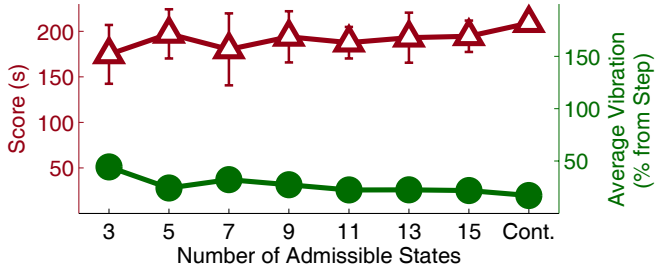
indicate the average payload swing throughout the trial. The vertical axes on the right side of the plots correspond to the payload swing data. These axes are in units of percent. One-hundred percent is equivalent to the oscillation caused by a step command of maximum velocity. The horizontal axis of these figures indicates the number of admissible states with which the trials were conducted. The data points associated with the scores are shown with error bars. These represent one standard deviation above and below the mean. Error bars for the vibration data are not shown because they are insignificantly small.



(a) ZV-Shaper Results.



(b) ZVD-Shaper Results.



(c) UMZV-Shaper Results.

Figure 5.9: Experimental Performance Results from the Crane Simulator.

It is evident from Figure 5.9 that the ZV, ZVD, and UMZV cases all exhibit improvements in score and vibration suppression as the number of admissible states is increased.

However, these improvements are most substantial for the ZV, and ZVD cases, especially as the number of admissible states changes from 3 to 9. After this point, performance generally continues to improve, but at a lower rate.

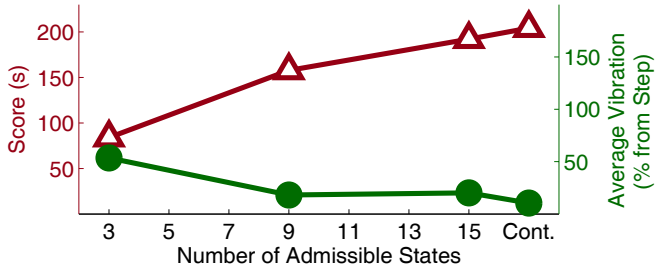
These results indicate that the detrimental effects of finite-state actuation on input shaping are more pronounced for course discretization than for fine discretization. Of course, this conclusion was anticipated. What is significant, however, is that the severity of these detrimental effects, for the ZV and ZVD cases, appear to diminish quickly in relation to the number of admissible states. Therefore, for cranes utilizing ZV or ZVD input shapers, these results suggests that relay-driven drives with a moderate number of actuation states can achieve similar system performance as more expensive variable-velocity-capable drives.

Another important feature of these graphs is that the range of scores and vibration levels for the case of the UMZV input shaper are noticeably more narrow than for either the ZV or ZVD cases. As the number of admissible states increases from 3 to continuous, the score for the UMZV case increases approximately 30 s from 170 to 200. Whereas for the ZV and ZVD cases, scores increase by approximately 100 s each. The high and low levels of vibration for the UMZV case ranges from approximately 45% to 5%, the ZVD case ranges from approximately 150% to 5%, and the ZV case ranges from approximately 70% to 10%.

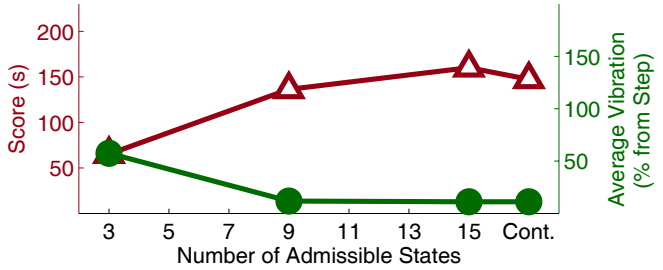
The disparity in performance range between the UMZV shaper and the positive impulse shapers indicates that, for this application, the UMZV shaper is affected least by finite actuation. This is significant in light of the fact that the UMZV case exhibited consistently high scores and low vibration throughout the range of admissible states. These results validate the performance assumptions stated in Section 5.1.2, which were based upon the $\mathbf{Q}_s$ distribution curves.

The score and vibration results obtained from the industrial crane are summarized in the plots of Figure 5.10. These plots were generated from a much smaller data set. In many cases, each data point represents a single trial, rather than the average of several trials. Consequently, no statistically significant error bars are shown with this data. Nevertheless, these results still exhibit characteristics that are similar to the large data set generated with

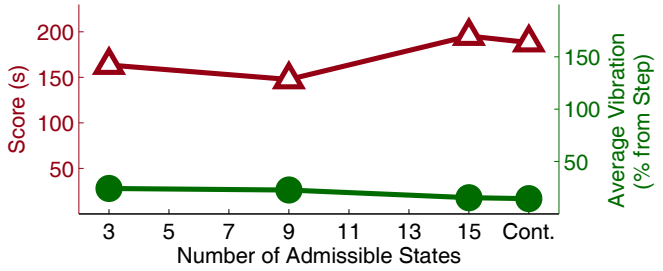
the crane simulator.



(a) ZV-Shaper Results.



(b) ZVD-Shaper Results.



(c) UMZV-Shaper Results.

Figure 5.10: Experimental Performance Results from the Industrial Crane.

5.3 Conclusion

Finite-state actuation of systems utilizing input-shaping filters can result in oscillation. The root cause of this undesirable behavior is that input-shaping filters can generate non-admissible actuation states. These commands are subsequently corrupted by the discretely actuated system, thereby reducing the vibration suppression capabilities of the input shaper. An analytical technique for identifying input-shaped states was presented. The operational effects of finite-state actuation were evaluated experimentally on an industrial crane and a crane simulator. It was demonstrated that a small increase in the number of actuation

states can result in an unproportionately large improvement in system performance. UMZV input shapers were affected less by finite-state actuation than ZV or ZVD input shapers.

CHAPTER VI

OVERCURRENTING: A SPECIAL CASE OF SATURATION[1]

CHAPTER SUMMARY: *Negative-impulse input shapers improve system rise-time beyond what is achievable using positive-impulse shapers. However, negative input shapers can cause overcurrenting, and subsequent system oscillation, when used with certain reference commands. This class of reference commands is examined in the context of the command space. The command space represents the space of all possible signals that may be issued to a system. It provides insight into how overcurrenting occurs, how overcurrenting can be mitigated, and the influence that mitigation strategies have on system performance. Two overcurrenting mitigation strategies are presented. The operational effects of overcurrenting and overcurrenting mitigation are evaluated using a 3-dimensional simulation of a bridge crane, and experimental results from a 10-ton industrial bridge crane.*

Figure 6.1 demonstrates the effectiveness of input shaping at reducing oscillation. The solid line is a velocity command that has been generated with an input-shaping filter. This command was used to actuate the trolley of the 10-ton bridge crane described in previous chapters of this thesis. Ordinarily, commanded motion causes large amplitude oscillations of the payload. However, in this case, the payload responded with very little transient or residual oscillation. This is shown with the dashed line of Figure 6.1.

An important property of the shaped velocity command shown in Figure 6.1 is that it remained between V_m and $-V_m$, the velocity limits of the crane. If a velocity command exceeds these limits, then the system clips the command so that it remains within the usable velocity range.

[1]Portions of this chapter are published in [103].

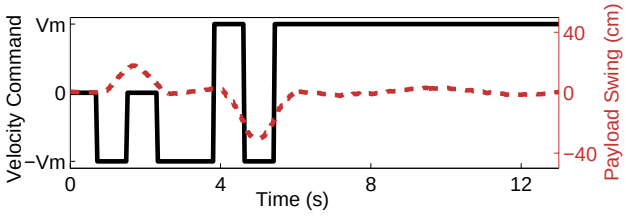


Figure 6.1: Shaped Command (Solid). Payload Response (Dashed).

Some input-shaped commands can exceed system actuator limits for short durations of time [80]. If the system cannot accommodate these brief periods of overcurrenting, then the shaped command is effectively corrupted, and the vibration-reducing properties of the command are degraded. This scenario was demonstrated by issuing a different shaped velocity command to the crane. This command is shown with the solid line of Figure 6.2. Note that this shaped command contains regions of overcurrenting that are unachievable by the crane. As a result, the shaped command is less effective at reducing oscillation than one remaining within the crane's velocity limits. Evidence of this is shown by the dashed line of Figure 6.2, where significant oscillation of the payload is exhibited.

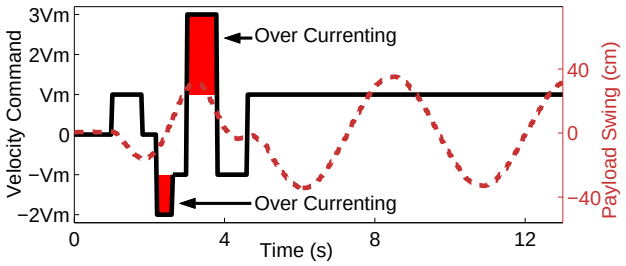


Figure 6.2: Shaped Command with Overcurrenting (Solid). Payload Response (Dashed).

The class of input-shaping filters that can cause overcurrenting and subsequent system oscillation is the subject of this chapter. Specifically, this chapter seeks to accomplish the following objectives:

1. *Identify conditions that cause overcurrenting.*

2. *Propose mitigation strategies to prevent overcurrenting.*

3. *Examine the influence that overcurrenting and overcurrenting mitigation have on the performance of human-operated systems.*

To accomplish these objectives, Section 6.1 presents some important properties of positive and negative input shapers. It is shown that negative input shapers, can give rise to overcurrenting under certain conditions. Section 6.2 identifies these conditions and presents mitigation strategies for preventing overcurrenting. The detrimental effects of overcurrenting, as well as the beneficial effects of the proposed mitigation techniques are demonstrated in Section 6.3 by using both simulated and experimental data from an industrial crane.

6.1 Properties of Positive and Negative Input Shapers

Recall that the set of all input shapers may be broadly categorized into two classes: positive input shapers, which are comprised entirely of positive impulses, and negative input shapers, which are permitted to contain negative, but finite valued, impulses.

Important properties of negative and positive input shapers may be ascertained by examining the time domain expression for a shaped command, $y_s(t)$. Recall that this expression is:

$$y_s(t) = A_1 x(t) + A_2 x(t - t_2) + \cdots + A_n x(t - t_n), \quad x(t < 0) = 0 \tag{6.1}$$

where $x(t)$ is the unshaped reference command, and A_i and t_i are the shaper impulse amplitudes and times, respectively.

From (6.1), the following inequality may be established:

$$|y_s(t)| \leq |A_1 x(t)| + |A_2 x(t - t_2)| + \cdots + |A_n x(t - t_n)| \tag{6.2}$$

If L is defined as $\max(|x(t)|)$, then it follows that:

$$|y_s(t)| \leq \Big[|A_1| + |A_2| + \cdots + |A_n|\Big] L \tag{6.3}$$

Equation (6.3) reveals that a shaped command can assume values that are greater than L, the maximum value of the original reference command. This is significant because often the limits imposed on a reference command reflect the limits inherent to the actuators of a

system. This means that if a shaped command is used to drive such a system, then actuator limits can be exceeded.

A necessary condition for the shaped command to exceed actuator limits is that the bracketed term in (6.3) must be greater than unity. Because (1.4) is enforced (the constraint that the impulse amplitudes must sum to unity), this condition is only possible when some of the impulses have negative values. Therefore, negative impulse input shapers are the only class of shaping filters that can cause overcurrenting to occur.

Although the use of negative input shapers can lead to overcurrenting, this filter class has some desirable characteristics. As discussed in Chapter 5, the negative UMZV shaper is less susceptible to the detrimental effects of finite-state actuation than the positive ZV or ZVD shapers. Negative input shaper durations can be significantly shorter than positive input shapers. Negative input shapers can also be used to produce on-off commands that are compatible with relay driven, or thruster-jet-type systems [50, 49, 79, 84, 83, 8]. The use of negative input shapers with on-off systems can also produce time-optimal or fuel-optimal commands [49, 79, 83]. Given these desirable attributes, it is evident that negative input shapers can provide useful control solutions provided that overcurrenting can be sufficiently controlled.

6.2 Negative Input Shaping and the Command Space

Systems that have upper and lower bounds on actuator effort cannot accommodate overcurrenting. The actuation process for this type of system is identical to that of saturation, and is schematically illustrated in Figure 6.3. To prevent overcurrenting when input shaping is not used, the reference command, $x(t)$, is constrained to be within a range defined by L:

$$|x(t)| \leq |L| \tag{6.4}$$

The saturation block in the diagram represents the nonlinear functionality of the system drives and motors. This block accepts shaped commands, and then actuates the system accordingly. However, commands exceeding the actuator limits of $\pm L$ are clipped.

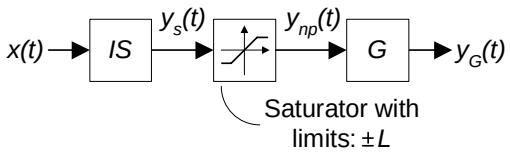


Figure 6.3: Limited Actuation Input Shaping.

Sections 6.2.1 and 6.2.2 examine the space of reference commands that cause negative input shapers to exhibit overcurrenting. Then, in Section 6.2.3, mitigation strategies are presented.

6.2.1 The Command Space

From (6.1) it is evident that the value of a shaped command at a given instant depends on both current and past values of the reference command. More specifically, for an n-impulse input shaper, $y_s(t)$ is a linear combination of $x(t)$, and $(n-1)$ past values of $x(t)$. This relationship can be expressed as the dot product of two vectors:

$$y_s(t) = \mathbf{A} \bullet \mathbf{X} \tag{6.5}$$

where

$$\mathbf{A} = \begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix} \tag{6.6}$$

$$\mathbf{X} = \begin{bmatrix} x(t) & x(t-t_2) & \cdots & x(t-t_n) \end{bmatrix} \tag{6.7}$$

The n-dimensional space containing the vector $\mathbf{X}$ is called the *command space*. The points contained within the command space represent all possible combinations of current and past values of $x(t)$. When n equals three, this space can be graphically represented as a cube, like the one shown in Figure 6.4. Because (6.4) holds, current and past values of $x(t)$ are bounded by $\pm L$, thus, each edge of the command-space cube has a length of $2L$.

Figure 6.5 helps one to visualize the physical significance of a point in the command space. This figure depicts a reference command, $x(t)$, "moving" forward in time. The portion of the command that has already been issued is shown with a dashed curve. The portion of the command that has not yet been issued is shown with a solid curve. A three-impulse input shaper requires three values of the reference command: $x(t)$, $x(t-t_2)$, and

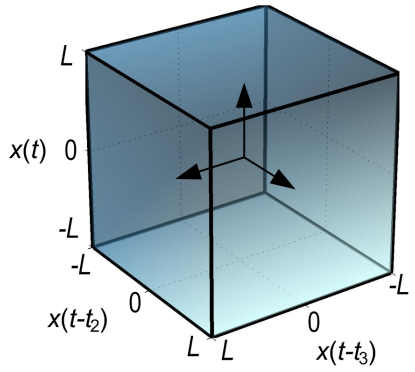


Figure 6.4: Command Space for a 3-Impulse Input Shaper.

$x(t - t_3)$. These values are the coordinates of a point, P, contained in a three-dimensional command space. The set of all such possible points comprises the entire space.

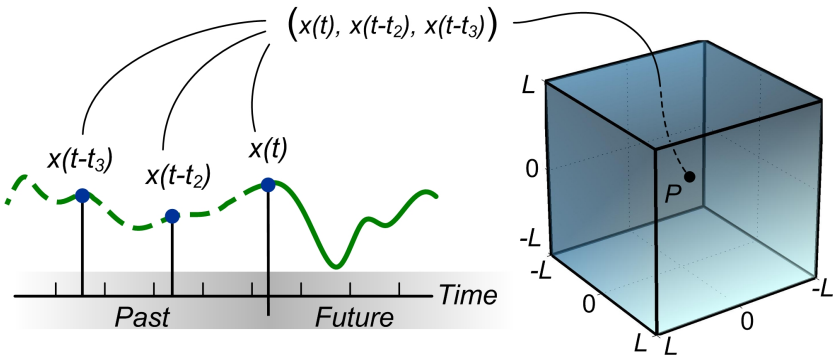


Figure 6.5: Current and Past Values of $x(t)$ Comprising the Coordinate of a Point in the Command Space.

6.2.2 Partitioning the Command Space

The command space may be partitioned into regions separated by planes. This partitioning will assist in determining when overcurrenting occurs. Recall that the equation of an $(n-1)$-dimensional plane in n-dimensional space is:

$$\alpha_1\chi_1 + \alpha_2\chi_2 + \cdots + \alpha_n\chi_n = \beta \tag{6.8}$$

where α_i is the coefficient of the variable, χ_i, and β is a constant.

Overcurrenting occurs when:

$$|y_s(t)| > L \tag{6.9}$$

The region of the command space that causes overcurrenting is determined by substituting (6.1) into (6.9). Two inequalities are obtained from this substitution:

$$A_1x(t) + A_2x(t - t_2) + \cdots + A_nx(t - t_n) > L \tag{6.10}$$

$$A_1x(t) + A_2x(t - t_2) + \cdots + A_nx(t - t_n) < -L \tag{6.11}$$

Comparing (6.8) with the first inequality reveals that (6.10) defines a plane that partitions the command space into two regions. The position and orientation of the plane is entirely determined by the coefficients of the input shaper, A_i, and the actuator limit, L. The region defined by the boundary of the command space and (6.10) contains all the points that cause negative input shapers to generate overcurrenting signals such that $y_s(t) > L$. Similarly, the inequality in (6.11) defines the region of points causing $y_s(t) < -L$.

An illustration of the partitioned command space when n equals three is shown in Figure 6.6. These partitions were generated for a UMZV input shaper. Reference commands corresponding to points within the outermost partitions cause overcurrenting. Actuators will saturate in response to these commands. Accordingly, these regions of overcurrenting are called *saturation regions*. Conversely, points in the innermost region correspond to reference commands that do not cause overcurrenting. This region is called the *drivable region*.

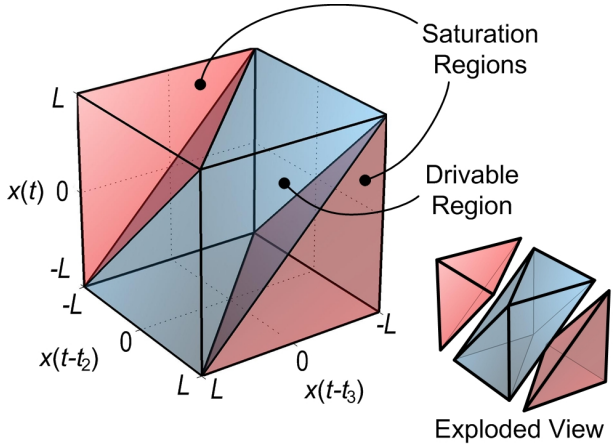


Figure 6.6: Saturated and Drivable Regions for a UMZV Input Shaper.

Interpreting negative input shapers in the context of a partitioned command space

provides the following insights. First, it is apparent that, generally speaking, reference commands that cause overcurrenting tend to switch quickly between states near $\pm L$. For example, a point in the upper-left-most corner of the command space indicates that the current value of the reference command is L. Previously, at time $t - t_2$, the value of the reference command was $-L$. Before that, at time $t - t_3$, the value of the reference command was L. Therefore, the reference command switched from L, to $-L$, and back to L within a time duration equal to t_3. This rapid "switching" behavior is characteristic of all points within the saturation regions. In contrast to rapidly switching reference commands, more benign reference commands can be described by points within the drivable region, and therefore will not cause overcurrenting.

The command space also reveals that the drivable region can be arbitrarily enlarged or reduced based upon the selection of the input shaper. This is because the boundaries of the command space partitions are defined by the shaper's impulse amplitudes. As a result, the space of admissible reference commands can be intelligently selected by choosing an appropriate input shaper. A class of negative input shapers has been developed that can serve this purpose. *Specified negativity* input shapers possess less aggressive impulse amplitudes than UMZV shapers, while still maintaining shorter durations than positive-impulse shapers [82]. Therefore, specified negativity shapers have a larger drivable region than UMZV shapers.

Finally, the existence of the drivable region suggests a remarkably simple overcurrenting mitigation strategy: constrain the reference commands in such a way that points in the command space that correspond to these commands are contained within the drivable region. This mitigation strategy is the focus of the next section.

6.2.3 Mitigation Strategies

Previous work on negative input shapers revealed that overcurrenting can occur [80, 82]. However, little has been published about mitigating this problem. Pao and Singhose proved that for a restricted class of reference commands, unity magnitude (UM) shapers will never cause overcurrenting [62] (UM shapers contain impulses whose amplitudes are constrained

to equal 1 or -1). The reference commands are restricted by imposing a state transition constraint.

6.2.3.1 State Transition Constraint for UM Shapers

The state transition constraint limits how quickly a reference command can transition from one value to another. If the time between transitions is greater than the duration of the input shaper, then overcurrenting will not occur [62]. This is because each transition is completely shaped before subsequent transitions begin.

Real-time implementation of this mitigation technique can be realized by using a digital sampler with a zero-order-hold. The mitigated system is illustrated with the block diagram shown in Figure 6.7. This system is identical to that shown previously in Figure 6.3, except that the reference signal, $x(t)$, is produced by sampling $u(t)$ at intervals of T_s. T_s must be greater than the duration of the UM shaper:

$$T_s > t_n \tag{6.12}$$

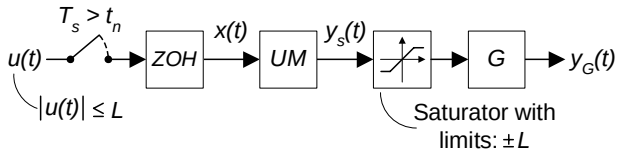


Figure 6.7: Mitigation of Overcurrenting Using the State Transition Constraint.

Equation (6.12) is called the *state transition constraint*. Enforcing (6.12) affects the relationship between current and past values of $x(t)$ in such a way that only points within a subset of the drivable region are utilized. To demonstrate this fact, consider the special case when a three-impulse UMZV shaper is used. As previously mentioned, implementation of the UMZV shaper requires the values of $x(t)$, $x(t-t_2)$, and $x(t-t_3)$. Enforcing (6.12) ensures that one of the following equalities related to these values holds at all times:

$$x(t) = x(t-t_2) \tag{6.13}$$

or

$$x(t-t_2) = x(t-t_3) \tag{6.14}$$

These equations may be written in standard plane-equation form as:

$$x(t) - x(t - t_2) + 0 \cdot x(t - t_3) = 0 \tag{6.15}$$

$$0 \cdot x(t) + x(t - t_2) - x(t - t_3) = 0 \tag{6.16}$$

When the plane surfaces defined by (6.15) and (6.16) are plotted within the command space, it is apparent that they are entirely contained within the drivable region. This relationship is illustrated in Figure 6.8. Plane A is the graphic representation of (6.16), and plane B is the graphic representation of (6.15). Thus, the state transition constraint guarantees that overcurrenting will not occur, because only a subset of points within the drivable region are utilized.

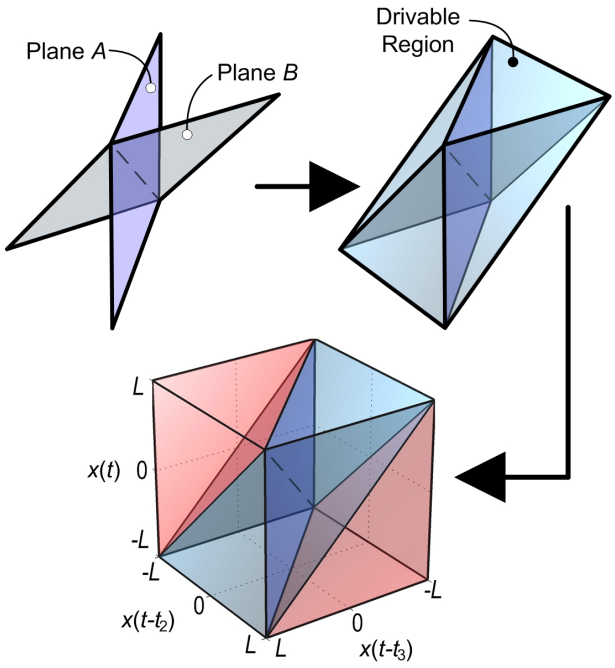


Figure 6.8: State Transition Constraint Represented as Two Planes in the Drivable Region of the Command Space.

The penalty associated with using the state transition constraint is that the space of available reference commands is significantly restricted. With the constraint, only two intersecting planes within the drivable region can be utilized. The operational effects of this limitation will be discussed in Section 6.3. But first, an alternative to the state transition

constraint called the *variable saturation constraint* is presented. This alternative to state transition constraint maximizes use of the drivable region. In contrast to the state transition constraint, which was developed for use with UM shapers, the variable saturation constraint can be utilized with all negative shapers.

6.2.3.2 Variable Saturation Constraint for Negative Shapers

Figure 6.9 illustrates how a variable saturation element can be used to prevent overcurrenting in systems with negative input shapers. A variable saturator, as opposed to a conventional static saturator, has threshold limits that can change in real-time.

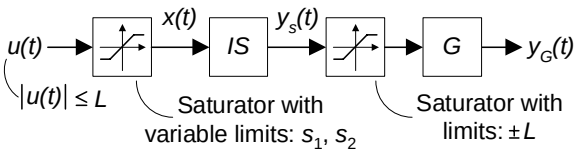


Figure 6.9: Mitigation of Overcurrenting Using the Variable Saturation Constraint.

To ensure that overcurrenting is prevented, the saturator must clip $u(t)$ so that shaped commands remain within the actuator limits:

$$|y_s(t)| \leq L \tag{6.17}$$

Two inequalities are obtained by substituting (6.1) into (6.17) yielding:

$$A_1x(t) + A_2x(t-t_2) + \cdots + A_nx(t-t_n) \leq L \tag{6.18}$$

$$A_1x(t) + A_2x(t-t_2) + \cdots + A_nx(t-t_n) \geq -L \tag{6.19}$$

These inequalities are the spacial complements of (6.10) and (6.11). Instead of defining the saturation regions, they define the drivable region through their intersecting domains and the boundary of the command space. Therefore, if (6.18) and (6.19) are satisfied for all time, then only the drivable region is utilized.

Equations (6.18) and (6.19) can be satisfied for all time by properly constraining $x(t)$. Appropriate constraints are obtained by solving (6.18) and (6.19) for $x(t)$:

$$s_1 \leq x(t) \leq s_2 \tag{6.20}$$

where

$$s_1 \equiv \frac{1}{A_1}[-L - A_2x(t-t_2) - \cdots - A_nx(t-t_n)] \tag{6.21}$$

$$s_2 \equiv \frac{1}{A_1}[L - A_2x(t-t_2) - \cdots - A_nx(t-t_n)] \tag{6.22}$$

Equation (6.20) is called the *variable saturation constraint*. It is enforced when s_1 and s_2 act as the lower and upper threshold limits of the variable saturator, respectively. Thus, the relationship between $x(t)$ and $u(t)$ is:

$$x(t) = \begin{cases} s_1 & , \ u(t) < s_1 \\ s_2 & , \ u(t) > s_2 \\ u(t), & \text{otherwise} \end{cases} \tag{6.23}$$

Therefore, enforcing (6.20) by using the variable saturator ensures that (6.18) and (6.19) hold for all time. Because unnecessary constraints are not imposed, the entire drivable region, as opposed to a restricted subset of the drivable region, is utilized. Consequently, overcurrenting is prevented and the maximum space of reference commands can be used.

6.3 Operational Effects of Negative Input Shaping

It has been shown that negative input shapers can cause overcurrenting and subsequent system oscillation. Mitigating strategies have been presented that prevent these problems, one using a subset of the drivable region, the other using the entire drivable region.

A shared attribute of both mitigation strategies is that reference commands are modified to prevent overcurrenting. In cases where the reference signal represents velocity or acceleration, the signal modification may prevent a system from achieving a desired final position or velocity, respectively.

Therefore, in light of the drivable region, and the reference command modification, several questions pertaining to the operational effects of overcurrenting and mitigation have yet to be addressed, specifically:

1. *Does reference command modification inhibit velocity or position trajectory tracking?*

2. *Does oscillation caused by overcurrenting detrimentally affect system performance?*

3. *Does the state transition constraint, which utilizes a subset of the drivable region, improve system performance?*

4. *Is performance further improved by using the variable saturation constraint, which utilizes the entire drivable region?*

The answers to these questions are application dependent. However, this section addresses these questions for one specific control application: human operation of a 10-ton bridge crane with UMZV input shaping. Given the results for this application, inferences about the performance of similar systems may be made.

6.3.1 Description of the Crane and Crane Simulation

The 10-ton bridge crane and crane simulator used during the experiments reported in Chapter 5 were also used to test the operational effects of overcurrenting and overcurrenting mitigation. 56 experimental trials from 11 volunteers were gathered by using the actual crane, 130 trials from 40 volunteers were gathered by using the simulation.

For the overcurrenting experiments, the crane and crane simulator were configured to accept continuously variable velocity commands within the admissible velocity range specified by $\pm L$. Two modes of manual control were enabled: a pendent, and a joystick. The pendent allows an operator to drive the crane in discrete directions and velocities that correspond to the different pendent button states (forward, reverse, left, and right). Pressing a pendent button issues a command to the crane to travel at full velocity in the specified direction. Two buttons may be pressed simultaneously in order to traverse diagonal trajectories. The joystick permits infinitely variable speed and direction control corresponding to the direction and magnitude that the joystick is manipulated.

Controlling the simulated crane is similar to controlling the industrial crane. A joystick-equipped computer allows the simulation to be manipulated exactly like the joystick-equipped industrial crane. Pendent control of the simulation is emulated with the arrow keys on a standard computer keyboard.

While the control of the actual and simulated cranes are very similar. Noticeable dynamic disparities do exist. First, the crane simulation behaves in accordance with linearized

and decoupled equations of motion. It is also capable of instantaneous acceleration and deceleration. The dynamic behavior of the actual crane does not reflect these simplifications. In spite of these dynamic differences, both the actual and simulated cranes are valuable tools because the gross effects of overcurrenting and overcurrenting mitigation can be visually discerned and easily measured.

6.3.2 Description of the Experiments

The skill level of the operators participating in the performance studies varied from novice to intermediate. Participants using the actual crane were required to undergo a short safety and crane operation training that included driving the crane to several locations. During the training, the crane was in manual mode, and did not have input shaping or overcurrenting mitigation enabled. Participants using the crane simulation did not undergo any special training.

Each experimental trial consisted of an operator driving the crane toward a target region. Every twenty seconds, the location of the target region changed to a different location within the workspace. The objective of the operator was to continually position the crane so that the payload remained within the moving target region for as long as possible. A complete trial consisted of 15 different target locations.

During each trial conducted on the 10-ton crane, the payload suspension cable remained fixed at a nominal operating length of 6 m, giving the system a period of approximately 5 s. The damping ratio was approximately 0.01. Each trial conducted on the crane simulator used a payload suspension cable length of 4 m and a damping ratio of 0.01, giving the simulated system a period of approximately 4 s.

The crane and crane simulator were equipped with a UMZV input shaper designed for the natural frequency and damping ratio of the system. Approximately one-third of the trials were conducted without mitigating overcurrenting, one-third of the trials were conducted while using the state transition constraint, and the final third were conducted while using the variable saturation constraint.

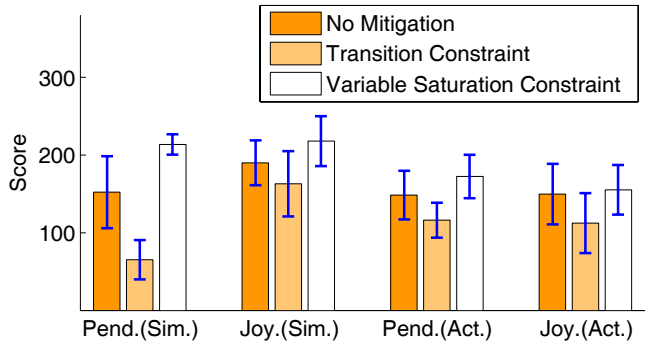
Several state variables were recorded during each experiment. These included the angular displacement of the payload, the position and velocity of the trolley, and the commanded velocity signals. These quantities were readily available from the simulation. But, for the industrial crane, a machine vision system, in conjunction with a Siemens programmable logic controller, were used to record these states.

6.3.3 Experimental Results

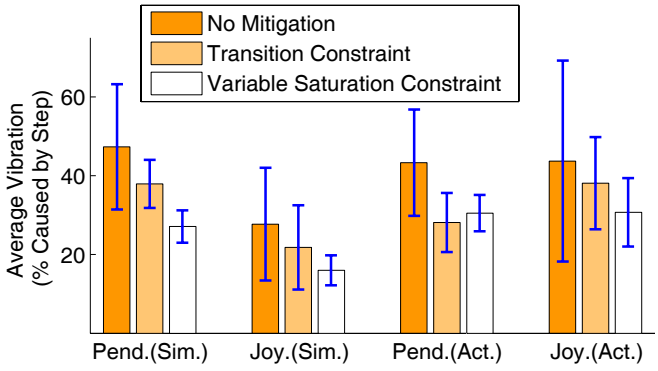
In order to assess the operational effects of overcurrenting and mitigation, two metrics were used to quantify system performance. The first was the time, in seconds, that the payload was contained within the target region. Higher values of this *score* indicate efficient crane operation, both in terms of gross trolley motion, and reduced payload sway. The average scores obtained by all the operators for each different testing configuration are shown with the bar graph in Figure 6.10(a). Error bars are also shown with this data that represent one standard deviation of the scores above and below the mean. One may observe from the data that in each control scenario, the transition constraint degraded operator performance, whereas the variable saturation constraint tended to improved operator performance.

The second metric used to gauge system behavior was the average angular displacement of the payload over the course of the trial[2]. Although there are several elements that contribute to total system oscillation, such as external disturbances and system nonlinearities, this metric still provides an indication of how much oscillation is caused by overcurrenting, and how the total oscillatory behavior of the system is affected by the mitigation techniques. The average angular displacements for each trial and testing configuration are shown with the bar graph in Figure 6.10(b). The vertical axis of this figure is in units of percent. One-hundred percent is equivalent to the oscillation caused by a step command of maximum velocity. Error bars are shown with the vibration data to indicate one standard deviation above and below the mean.

[2]It should be noted that this measurement differs fundamentally from average *residual* angular displacement. Residual oscillation refers to payload sway exhibited after a command has reached a steady-state value. Total oscillation includes payload sway during transient and steady-state portions of a command.



(a) Average Score.



(b) Average Angular Displacement of the Payload.

Figure 6.10: Experimental Performance Results.

Figure 6.10(b) indicates that payload oscillation can be noticeably reduced when overcurrenting is mitigated. This result validates the effectiveness of the proposed mitigation strategies at preventing overcurrenting and subsequent oscillation. This result also indicates that reference signals generated by human crane operators utilize saturation regions of the command space, and therefore cause overcurrenting.

Considering the results of Figure 6.10(a) in light of Figure 6.10(b) reveals how use of the command space can affect system performance. Figure 6.10(a) indicates that the average score while using the state transition constraint was lower than the average score when no mitigation was used. Therefore, although the constraint reduced overcurrenting-induced oscillation, the constraint also inhibited efficient operation because only a small subset of the drivable region was utilized. Conversely, the variable saturation constraint utilizes the entire drivable region. As a result, overcurrenting was prevented while use of the command

space was maximized. This is demonstrated by the high scores for the saturation constraint in Figure 6.10(a), and low oscillation results in Figure 6.10(b).

6.4 Conclusion

Negative input shapers can cause periods of overcurrenting that may result in significant oscillation. The command space representation of reference signals has been developed to classify the types of signals that cause overcurrenting. This space provides insight into how overcurrenting occurs, how it can be mitigated, and the influence that different mitigation strategies have on system performance. Points within the command space called the drivable region correspond to reference commands that do not cause overcurrenting. Two mitigation strategies have been presented that prevent overcurrenting by constraining reference commands to be within this region. One of these methods utilizes the entire drivable region, while the other utilizes only a small subset of this region. Experiments on a 10-ton industrial bridge crane, and a 3-dimensional simulation of this crane, validate the effectiveness of the mitigation strategies. It has been demonstrated that significant performance benefits are gained by maximizing the use of the drivable region, while preventing overcurrenting.

CHAPTER VII

A VISUAL TOUCHSCREEN INTERFACE FOR CRANE MANIPULATION[1]

CHAPTER SUMMARY: *The role of the human operator in crane manipulation is multifaceted. Generally, it involves combining skilled manual control with supervisory tasks such as monitoring throughput, assessing potential hazards, and payload transport decision making. This chapter investigates these tasks from the perspective of fundamental human factors theory. Models of human/crane interaction are presented to motivate change in the way that operators interact with cranes. A new kind of operator interface is presented, which simplifies crane manipulation. A thorough description of this interface is presented.*

Bridge and gantry cranes occupy a crucial role in industry. They are used throughout the world in thousands of shipping yards, construction sites, steel mills, warehouses, nuclear power and waste storage facilities, and other industrial complexes. The efficiency and safety of cranes are important contributors to industrial productivity. For this reason, improving the operational effectiveness of cranes can be extremely valuable.

Efficient and safe crane operation is inhibited largely by hook and payload swing. Consequently, tremendous efforts have been directed toward the goal of improving crane performance by developing and implementing various dynamic-control schemes. These techniques are specifically intended to reduce payload and hook swing caused by external disturbances and/or issued commands. Some of these dynamic control schemes have proven to be both effective at reducing unwanted oscillation and simple to implement on industrial cranes [76,86,108,85]. Operators that manipulate cranes utilizing appropriate dynamic compensation can generate safer and more efficient motions than operators that manipulate

[1]Portions of this chapter are published in [109].

cranes without dynamic compensation [40, 85].

Although significant strides have been made to improve safety and efficiency by controlling a crane's dynamic response to issued commands, relatively little consideration has been given to the way in which operators issue those commands to a crane. Conventional crane operation is accomplished in much the same manner as in decades past - through the use of a control pendent or actuation levers. Examples of these interface devices are shown in Figure 7.1. A pendent permits a crane to be driven in discrete directions and velocities that correspond to the different button states. A lever interface is similar, each lever controls exactly one mode of actuation (i.e. forward-reverse, left-right, up-down).

(a) Lever Interface by Magnetek. (b) Pendent Interface by Magnetek.

Figure 7.1: Conventional Crane Interface Devices.

A new interface has been developed that permits operators to command crane motion by touching a video monitor. The monitor displays real-time images of the crane and its workspace. An operator specifies a desired hook destination by simply touching the image of the destination displayed on the video monitor. This visual and touchscreen interface scheme is utilized in conjunction with a feedback control system that drives the crane to the target location. The objective in developing the new interface is to improve the efficiency and safety of crane operation during positioning tasks. The system also provides the means by which to conduct remote manipulation.

This chapter presents both a description of, and motivation for, the new crane interface. Specifically, this chapter seeks to:

1. *Use established human factors theory to develop models of human/crane interaction.*

2. *Use the developed human/crane interaction models to motivate the development of the new crane interface.*

3. *Provide a thorough description of the mathematics that enable the interface to function.*

The following section applies established human factors theory to the special case of crane control in order to develop models of human/crane interaction. Section 7.2 then provides a thorough description of the interface, as well as a mathematical description of how two-dimensional coordinates from the video monitor are mapped to the three-dimensional coordinates of the crane-workspace. Section 7.2 also addresses the positioning accuracy achievable with the interface.

7.1 Human Factors in Crane Control

Significant strides have been made toward the objective of enhancing crane operability, specifically in the areas of safety, efficiency, and positioning accuracy. However, these advances have been primarily concerned with controlling the dynamic response of cranes.

Several researchers have investigated open-loop control techniques to suppress residual command-induced payload swing [76, 86, 1, 59]. Extensions of this work have resulted in open-loop methods that also reduce transient payload swing [81, 29, 46, 72]. A tremendous number of closed-loop techniques have also been developed. For example, suppression of disturbance-induced payload swing was achieved on a 10-ton bridge crane that used on-off, relay-type motor actuation [27]. A bridge crane capable of executing variable velocity commands was used to test a combined input-shaping and proportional-derivative control architecture [108]. The control enabled the crane to achieve precise hook positions while rejecting both motion and disturbance-induced oscillations. Similar objectives were achieved in [64] using feedback linearization and hoist control. Several other closed-loop control schemes have also been developed that exclusively address crane dynamics [19, 56, 68, 59, 44, 99].

The chief distinction between the prior work and the system developed here is the process level affected by the innovation. That is, the system developed here affects how an operator

issues commands to a crane; the prior work is concerned with a crane's response to issued commands.

This distinction is more clearly defined in Sheridan's supervisory control model of an automation system [75]. In Figure 7.2, this construct has been adapted specifically for cranes, where the human operator and crane are linked through a computerized process control. This model describes different modes of interaction and information exchange between the human operator and the crane. Specifically, the modes of interaction are:

a. Operator specifies crane motion indirectly through a human/machine interface.

b. Operator receives the state of the crane indirectly through the computer.

c. Crane receives trajectory commands from the computer.

d. Operator intervenes directly in the process.

e. Operator directly observes the process.

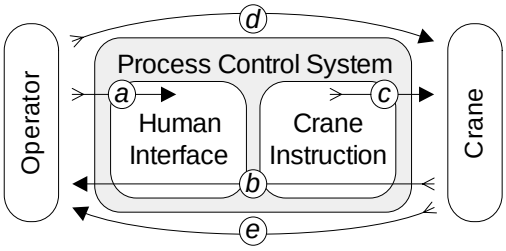


Figure 7.2: Supervisory Control Model.

The process control system may be considered as two interconnected processes: the crane instruction process that directly controls the crane motion, and the human interface process that communicates with the operator [113]. The bulk of research directed at improving crane performance has done so by implementing various control algorithms at the crane instruction process level. The instructions generated by these algorithms are transmitted through interaction mode *c*. In contrast, the innovation developed here is implemented at the human interface process level. It directly affects the information transmitted through interaction mode *a*.

7.1.1 Operator Tasks and Behavior

To understand the potential benefits derived by addressing the interface of a crane system, one can consider the tasks that operators perform during a transport and hoisting operation and how these tasks relate to operators' cognitive behaviors.

Stahre subdivided operator tasks for generalized automation processes [113]. Here, Stahre's divisions are listed in the context of crane control:

- **Planning** involves gaining pertinent information about the crane, tasks, and surroundings, then determining how the crane will be used to accomplish specific manipulation tasks. Planning also involves selecting an appropriate trajectory.

- **Commanding** involves communicating the desired motion to the crane. The methods of communication can vary from depressing pendent buttons to using abstract high-level programming languages.

- **Monitoring** the motion of the crane forms a closed control loop. In this normal state of operation, the operator is constantly acquiring, estimating, and evaluating process information. The information is used to draw conclusions about the manipulation task. Actions resulting from monitoring usually involve minor adjustments to crane trajectories.

- **Intervening** is a conditional task triggered by a malfunction or other unanticipated event (e.g. an operator detects that a collision is imminent, and intervenes by pressing an emergency stop button).

- **Learning** is required when an unknown error occurs. The analysis of the cumulative information gathered by the operator is used to resolve the error.

According to fundamental theory of human/machine systems, each of the operator tasks invokes different cognitive behaviors [70]. In order of increasing complexity and cognitive resource use, these behaviors are:

- **Intuition-based behavior:** instinctual sensory-motor behavior, analogous to everyday trivial tasks such as picking up a spoon while eating breakfast.

- **Rule-based behavior:** actions triggered by a certain pattern of stimuli. These actions could be imitated by a computer using "if-then" statements to initiate an appropriate response.

- **Critical thinking-based behavior:** making complex decisions and multi-factor scheduling of actions.

An informative and descriptive framework for evaluating the crane operation process is through the use of the cognitive activity demand matrix, developed by Hollnagel [28]. This tool combines operator tasks and cognitive behaviors in a matrix format that clarifies how the cognitive behaviors of crane operators are related to the various tasks they perform.

The matrix shown in Table 7.1 represents the crane manipulation process when operators interface with the system through a control pendent or actuating levers. When an operator attempts to command crane motion, he or she mentally accomplishes several steps. First, the desired trajectory created during the planning task is decoupled into the kinematic components corresponding to the different modes of actuation (i.e. forward-reverse, left-right, up-down). Second, the decoupled trajectories are mentally mapped to the corresponding pendent buttons or actuating levers. Finally, the operator attempts to physically actuate the correct combination and sequencing of pendent buttons or levers. Because this three-step process can be quite complex and time consuming for anything but the simplest of trajectories, operators often formulate trajectories requiring motion in only one degree-of-freedom at a time.

An important aspect of this type of operator behavior is revealed in the evaluation matrix. Namely, commanding crane motion through a pendent or lever-type interface requires extensive use of the higher rule and thinking-based cognitions.

The motivation for developing a new interface is rooted in minimizing the cognitive complexity and resource use operators experience during the commanding activity. A tremendous benefit of "shifting" operator actions toward intuition-based behavior is that more of

Table 7.1: Evaluation Matrix for Pendent or Lever Crane Actuation.

Increasing Cognitive Complexity & Resource Use

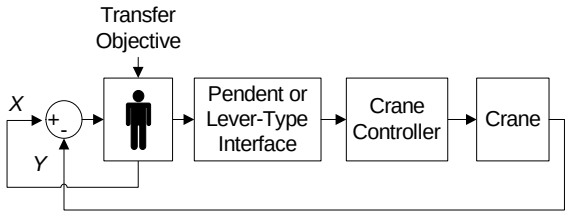
		Cognitive Behavior		
		Intuition-Based	Rule-Based	Thinking-Based
Operator Tasks	Plan			✓
	Command		✓	✓
	Monitor	✓		
	Intervene		✓	
	Learn			✓

their mental resources are free to meet new situations and monitor existing states of the system [113].

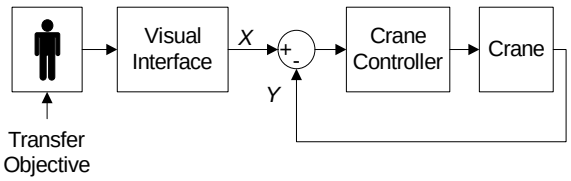
The visual touchscreen interface developed here aids in shifting mental processes toward simplified cognitive behavior by simplifying the commanding task. Conventional pendent or lever-type interfaces require operators to decouple desired trajectories, map decoupled components to pendent buttons or levers, and schedule actuating events - tasks that utilize rule-based and thinking-based behaviors. With the visual touchscreen interface, an operator commands crane motion by touching points along a desired path; this type of gesture-based control utilizes intuition-based behavior [3, 21].

The block diagrams in Figure 7.3 provide additional insight into how the visual touchscreen interface simplifies crane operation when compared with traditional interface devices. These block diagrams represent the motion control process of a crane using a pendent or lever-type interface (Figure 7.3(a)) and the visual touchscreen interface (Figure 7.3(b)). In both scenarios, an operator decides upon a desired end state of the crane, X, based upon some transfer objective. If the operator is using the visual interface, then the end state is input into the system through the intuition-based behavior of touching a video monitor. At this point, the commanding task is completed, and the operator may proceed to the monitoring task. In contrast, when an operator uses a conventional interface, the desired end state is continually compared to the actual crane state, Y, with the operator acting as a control element in a feedback loop. Based upon the continuous comparison between actual and desired crane states, the operator proceeds to the more mentally intensive tasks

discussed earlier, path planning, decoupling, mapping, and actuation scheduling.



(a) Pendent or Lever-Type Interface.



(b) Visual Touchscreen Interface.

Figure 7.3: Crane Motion Control Using Conventional and Visual Interfaces.

This type of in-the-loop interaction contributes most significantly toward performance differences between novice and expert operators [22]. Highly skilled operators have a very refined *in-the-loop* operational ability, whereas novice operators lack skill in this area. In contrast, both novice and expert operators possess comparable capacities to plan motion, monitor, intervene, and learn. Therefore, because the visual interface eliminates the main distinguishing factor between expert and novice operators, one may expect their respective performances to be similar.

In light of the cognitive-behavior and operator-in-the-loop crane manipulation models, three results are anticipated by implementing the visual interface: First, operators will no longer be impelled to formulate path plans that use only one degree-of-freedom at a time. That is, instead of operators generating path plans that are kinematically simple, spatially simple trajectories can be automatically generated by the controller. This will result in shorter, more efficient crane motions. Second, operators will have an increased capacity for monitoring the state of the system, resulting in an increased awareness of safety. Third, the performance gap between novice and highly skilled operators will be narrowed. Chapter 8 presents the results of several operator studies involving the visual touchscreen interface that support these assertions.

7.2 Interface Description

Figure 7.4 shows an illustration of a visual touchscreen interface integrated into a bridge crane work environment. A downward looking camera provides a live video stream of the crane and its workspace, which is displayed on a touchscreen monitor.

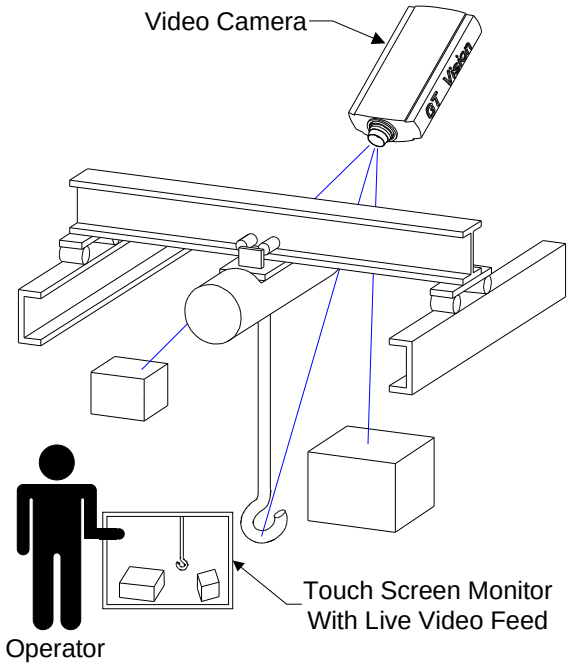


Figure 7.4: Overview of a Vision Interface Integrated in a Bridge Crane Work Environment.

An operator specifies a desired 3-dimensional hook destination by touching the 2-dimensional image of the desired destination shown on the monitor. Because 3-dimensional information is being extracted from a 2-dimensional image, additional information is needed. Therefore, the operator also specifies the height of the desired destination that was touched. In many cases, the object touched is a point on the floor of the workspace, and thus the height is zero. Other times, it is the top of an item that will be hoisted, and thus its height is known or easily estimated.

Once the desired destination is completely specified, then the operator issues a "GO" command. The computer controller then automatically drives the crane to the destination. An operator may also optionally specify a minimum travel height in order to avoid obstacles. This method of crane manipulation can be used in conjunction with dynamic compensation

to prevent motion and disturbance-induced oscillation.

Figure 7.5 is a screen shot of the visual interface used to manipulate a 10-ton bridge crane at the Georgia Institute of Technology (this crane was described previously in Section 3.1.4). In region A, the user touches the desired hook destination. Region B and C are used to enter object height, and travel height, respectively.

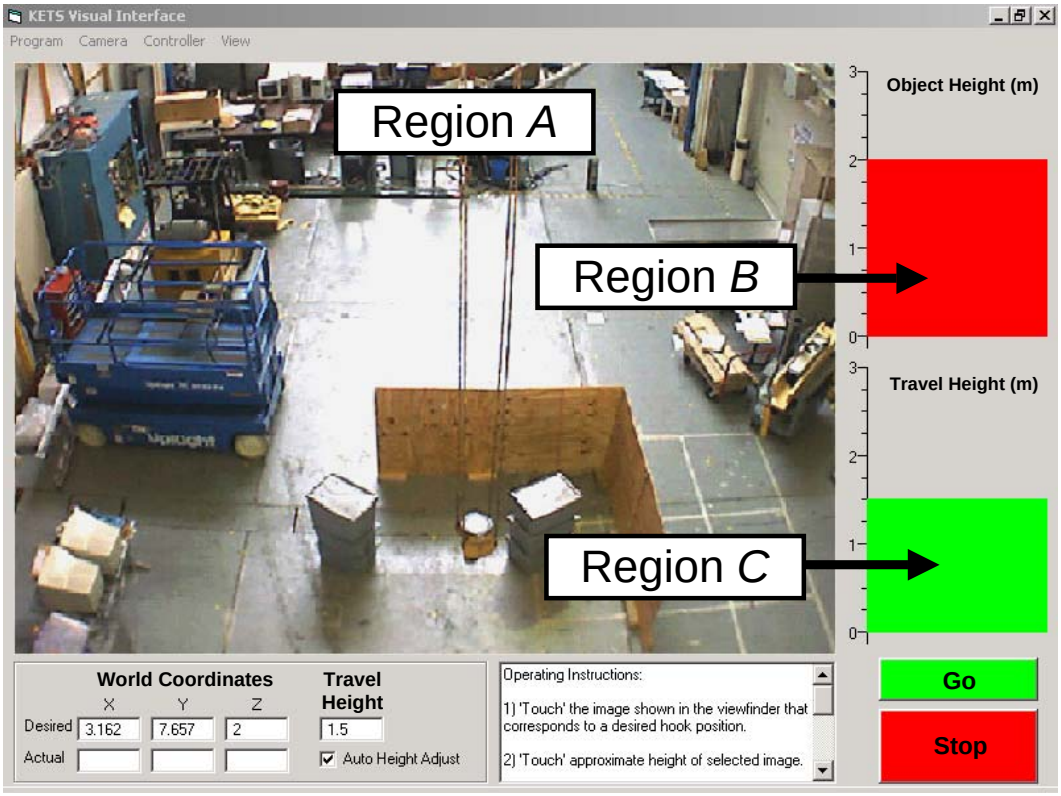


Figure 7.5: Visual Touchscreen Interface.

This method of commanding crane motion poses a fundamental implementation challenge: pixel information from a two-dimensional image must be converted into a desired three-dimensional position in the crane workspace. Mathematical relationships that provide this functionality for an ideal camera (i.e. no image distortion) are described in Section 7.2.1 where the position of the camera lens, as well as the position and orientation of the camera's imaging sensor (CCD) are assumed to be known. Section 7.2.2 describes a calibration method for determining these assumed quantities. Section 7.2.3 explains how the mathematical relationships established in Sections 7.2.1 and 7.2.2 can be applied to non-ideal cameras exhibiting image distortion due to spherical aberration and misalignment of

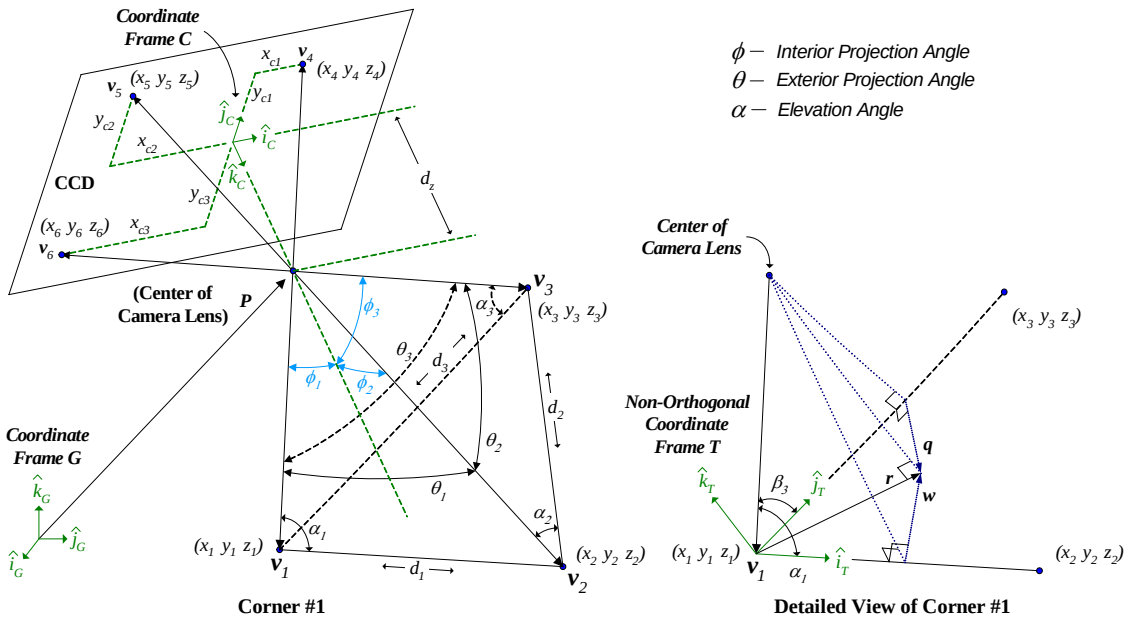


Figure 7.6: Geometric Relationships Between the CCD, Camera Lens, and Points in Workspace.

the camera lens and CCD. Finally, Section 7.2.4 experimentally evaluates the precision with which workspace coordinates can be specified with the visual interface. These sections may be bypassed without loss of continuity, but they are provided for the reader interested in knowing how the visual interface was implemented.

The geometric relationships necessary for understanding the mathematical developments are shown in Figure 7.6. The leftmost portion of this figure is an illustrative overview geometrically relating the CCD, camera lens, and points in the surrounding workspace. The rightmost portion of the figure provides added detail to the drawing. The parameters used in this illustration are defined as follows:

$\widehat{\boldsymbol{i}}_G$, $\widehat{\boldsymbol{j}}_G$, $\widehat{\boldsymbol{k}}_G$	Orthogonal unit vectors fixed in the crane workspace coordinate system, frame G.
$\boldsymbol{P}$	Position vector originating at the origin of the workspace coordinate system and terminating at the geometric center of the camera lens.
$\widehat{\boldsymbol{i}}_c$, $\widehat{\boldsymbol{j}}_c$, $\widehat{\boldsymbol{k}}_c$	Orthogonal unit vectors of frame C, the coordinate system affixed to the CCD. $\widehat{\boldsymbol{k}}_c$ is collinear with the line passing through the origin of C and the geometric center of the camera lens.
x_i, y_i, z_i	Coordinates of points #1, #2, and #3, relative to frame G.

$\widehat{\boldsymbol{i}}_T$, $\widehat{\boldsymbol{j}}_T$, $\widehat{\boldsymbol{k}}_T$	Unit vectors of the coordinate system affixed to corner #1. Vectors $\widehat{\boldsymbol{i}}_T$ and $\widehat{\boldsymbol{j}}_T$ are defined to be collinear with the lines connecting points 1 and 2, and points 1 and 3, respectively. Vector $\widehat{\boldsymbol{k}}_T$ is perpendicular to the plane containing $\widehat{\boldsymbol{i}}_T$ and $\widehat{\boldsymbol{j}}_T$.
$\boldsymbol{v}_i$	Position vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ originate at the camera lens and terminate at the corresponding x_i, y_i, and z_i points. Position vectors $\boldsymbol{v}_4$, $\boldsymbol{v}_5$, and $\boldsymbol{v}_6$ are, collinear with $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$; they originate at the camera lens and terminate at the CCD.
x_{ci}, y_{ci}	Coordinates, relative to frame C, of the points where $\boldsymbol{v}_4$, $\boldsymbol{v}_5$, and $\boldsymbol{v}_6$ intersect with the plane defined by the CCD.
$\boldsymbol{r}$	Projection of $\boldsymbol{v}_1$ onto the plane containing $\widehat{\boldsymbol{i}}_T$ and $\widehat{\boldsymbol{j}}_T$.
$\boldsymbol{w}$, $\boldsymbol{q}$	Vectors in the plane containing $\widehat{\boldsymbol{i}}_T$ and $\widehat{\boldsymbol{j}}_T$.
d_z	Focal length of the camera lens (distance from the origin of frame C to the camera lens).
ϕ_i	Interior projection angles (angle between $\boldsymbol{v}_i$ and $\widehat{\boldsymbol{k}}_c$).
θ_1, θ_2, θ_3	Exterior projection angles (angle between $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$; $\boldsymbol{v}_2$ and $\boldsymbol{v}_3$; $\boldsymbol{v}_3$ and $\boldsymbol{v}_1$, respectively).
α_1, α_2, α_3	Elevation angles (angle between $\boldsymbol{v}_1$ and $\boldsymbol{v}_1 - \boldsymbol{v}_2$; $\boldsymbol{v}_2$ and $\boldsymbol{v}_2 - \boldsymbol{v}_3$; and $\boldsymbol{v}_3$ and $\boldsymbol{v}_3 - \boldsymbol{v}_1$, respectively).
β_3	Angle between $\boldsymbol{v}_1$ and $\boldsymbol{v}_1 - \boldsymbol{v}_3$ ($\beta_3 = \pi - \theta_3 - \alpha_3$).
d_1, d_2, d_3	Distance between various points in the workspace.

7.2.1 Obtaining Object Coordinates from Image

Given the position and orientation of frames C and G, the coordinate frames affixed to the CCD and crane workspace, respectively, and d_z, the focal length of the camera lens, the coordinate of a point in the crane workspace (x, y, z) can be determined in parametric form from the coordinate of a pixel on the CCD (x_c, y_c) by:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} + {}^{G}_{c}R \begin{bmatrix} -x_c \\ -y_c \\ d_z \end{bmatrix} t \tag{7.1}$$

where ${}^{G}_{c}R$ is the rotation matrix relating the orientation of frame C to that of frame G, and t is a parametric length operator. P_x, P_y, and P_z are the components of $\boldsymbol{P}$ relative to frame G. The coordinate pair (x_c, y_c) is specified by the operator when the object image is touched. This expression specifies a point along a ray with length t emitting from the camera lens and directed toward the object specified by the operator. The point at which the ray intersects the object is the three-dimensional coordinate of the two-dimensional image.

To explicitly solve for (x, y, z), either x, y, z, or t must be known *a priori*. As discussed previously, usually z is known, and corresponds to the height of the desired destination touched[2].

When t is eliminated from (7.1) by specifying z, then x and y can be determined from:

$$x = P_x + a\frac{z - P_z}{c} \tag{7.2}$$

$$y = P_y + b\frac{z - P_z}{c} \tag{7.3}$$

where,

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = {}^{G}_{c}R \begin{bmatrix} -x_c \\ -y_c \\ d_z \end{bmatrix} \tag{7.4}$$

7.2.2 Position and Orientation Calibration

Prior to the video camera being used in the manner described in Section 7.2.1, both its position and orientation relative to the fixed frame G must be determined. There are several methods for obtaining these quantities, including direct measurement. Although manual measurement of the camera's position can be accomplished with some degree of accuracy, directly measuring the orientation of the camera is difficult and prone to errors.

To eliminate manual measurements, an automatic calibration method is presented here. Automatic calibration can be quickly performed by an operator once the mathematical

[2]The coordinate z corresponds to the height of a touched object when the fixed coordinate frame G is defined with the $\widehat{\boldsymbol{i}}_G$ - $\widehat{\boldsymbol{j}}_G$ plane coincident with the workspace floor.

calibration algorithms are programmed into a computer. The only action required by the operator to perform the calibration is that he or she must use the touchscreen interface to select three points in the workspace, then the coordinates of these points, (x_1, y_1, z_1), (x_2, y_2, z_2), and (x_3, y_3, z_3), must be entered into the computer.

Rays emitting from the three workspace points that pass through the camera lens intersect the CCD at the known points (x_{c1}, y_{c1}), (x_{c2}, y_{c2}), and (x_{c3}, y_{c3}). These coordinates comprise the x and y components (relative to frame C) of vectors $\boldsymbol{v}_4$, $\boldsymbol{v}_5$, and $\boldsymbol{v}_6$, respectively. The z component of these vectors is d_z, the known focal length of the camera lens. Therefore, the exterior projection angles, θ_i, may be obtained from:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \cos^{-1} \begin{bmatrix} \frac{\boldsymbol{v}_4 \bullet \boldsymbol{v}_5}{|\boldsymbol{v}_4||\boldsymbol{v}_5|} \\ \frac{\boldsymbol{v}_5 \bullet \boldsymbol{v}_6}{|\boldsymbol{v}_5||\boldsymbol{v}_6|} \\ \frac{\boldsymbol{v}_6 \bullet \boldsymbol{v}_4}{|\boldsymbol{v}_6||\boldsymbol{v}_4|} \end{bmatrix} \tag{7.5}$$

The strategy for determining the lens position, denoted by the vector, $\boldsymbol{P}$, uses the known exterior projection angles from (7.5) and the known workspace points to calculate the elevation angles, α. From these, the position of the geometric center of the lens easily follows.

Given the exterior projection angles, θ_i, from (7.5), the distances, d_i, between the known workspace points, and the law of sines, the following relationships can be stated:

$$\begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} \equiv \begin{bmatrix} \frac{d_1}{\sin(\theta_1)} \\ \frac{d_2}{\sin(\theta_2)} \\ \frac{d_3}{\sin(\theta_3)} \end{bmatrix} = \begin{bmatrix} \frac{|\boldsymbol{v}_2|}{\sin(\alpha_1)} \\ \frac{|\boldsymbol{v}_3|}{\sin(\alpha_2)} \\ \frac{|\boldsymbol{v}_1|}{\sin(\alpha_3)} \end{bmatrix} = \begin{bmatrix} \frac{|\boldsymbol{v}_1|}{\sin(\pi-\theta_1-\alpha_1)} \\ \frac{|\boldsymbol{v}_2|}{\sin(\pi-\theta_2-\alpha_2)} \\ \frac{|\boldsymbol{v}_3|}{\sin(\pi-\theta_3-\alpha_3)} \end{bmatrix} \tag{7.6}$$

The absolute values $|\boldsymbol{v}_1|$, $|\boldsymbol{v}_2|$, and $|\boldsymbol{v}_3|$ may be eliminated from (7.6) resulting in the following system of equations:

$$k_1 \sin(\theta_1 + \alpha_1) - k_3 \sin(\alpha_3) = 0 \tag{7.7}$$

$$k_2 \sin(\theta_2 + \alpha_2) - k_1 \sin(\alpha_1) = 0 \tag{7.8}$$

$$k_3 \sin(\theta_3 + \alpha_3) - k_2 \sin(\alpha_2) = 0 \tag{7.9}$$

Solving (7.8) and (7.9) for α_2 and α_3 yields:

$$\alpha_2 = \sin^{-1}\left(\frac{k_1 \sin(\alpha_1)}{k_2}\right) - \theta_2 \tag{7.10}$$

$$\alpha_3 = \sin^{-1}\left(\frac{k_2 \sin(\alpha_2)}{k_3}\right) - \theta_3 \tag{7.11}$$

Combining (7.10) and (7.11), then substituting the result into (7.7) yields:

$$k_1 \sin(\theta_1 + \alpha_1) - k_3 \sin\left\{\sin^{-1}\left[\frac{k_2}{k_3}\sin\left(\sin^{-1}\left[\frac{k_1 \sin(\alpha_1)}{k_2}\right] - \theta_2\right)\right] - \theta_3\right\} = 0 \tag{7.12}$$

Equation (7.12) can be solved numerically for α_1. The solution yields at most four possible values of α_1 due to the multiple arcsine terms in the expression. Substituting the resulting α_1 values into (7.10) yields at most eight possible values of α_2 . Finally, α_2 may be substituted into (7.11) to yield at most sixteen possible values of α_3. In total, the numerical solution to the system of equations in (7.7) - (7.9) provides at most sixteen possible combinations of α_1, α_2, and α_3. In practice, however, most of these combinations may be immediately eliminated because they are redundant or contain imaginary components. Using the remaining combinations of elevation angles, the possible magnitudes of vectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ are easily calculated from (7.6).

From the detailed view of corner 1 in Figure 7.6, one observes that:

$$\boldsymbol{r} = |\boldsymbol{v}_1| \cos(\alpha_1)\widehat{\boldsymbol{i}}_T + |\boldsymbol{w}|\left(\widehat{\boldsymbol{k}}_T \times \widehat{\boldsymbol{i}}_T\right) \tag{7.13}$$

$$\boldsymbol{r} = |\boldsymbol{v}_1| \cos(\beta_3)\widehat{\boldsymbol{j}}_T + |\boldsymbol{q}|\left(\widehat{\boldsymbol{j}}_T \times \widehat{\boldsymbol{k}}_T\right) \tag{7.14}$$

Defining the components of the unit vectors comprising frame T as:

$$\begin{bmatrix} \widehat{\boldsymbol{i}}_T \\ \widehat{\boldsymbol{j}}_T \\ \widehat{\boldsymbol{k}}_T \end{bmatrix} = \begin{bmatrix} i_{Tx} & i_{Ty} & i_{Tz} \\ j_{Tx} & j_{Ty} & j_{Tz} \\ k_{Tx} & k_{Ty} & k_{Tz} \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} \tag{7.15}$$

and the simplifying terms g_i, and h_i as:

$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ -\widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \widehat{\boldsymbol{k}}_T \times \widehat{\boldsymbol{i}}_T \tag{7.16}$$

$$\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ -\widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \widehat{\boldsymbol{j}}_T \times \widehat{\boldsymbol{k}}_T \tag{7.17}$$

equations (7.13) and (7.14) can be equated to solve for the magnitudes of $\boldsymbol{w}$ and $\boldsymbol{q}$ as:

$$|\boldsymbol{w}| = \frac{1}{g_1}\Big(|\boldsymbol{v}_1|\cos(\beta_3)j_{Tx} + |\boldsymbol{q}|h_1 - |\boldsymbol{v}_1|\cos(\alpha_1)i_{Tx}\Big) \tag{7.18}$$

$$|\boldsymbol{q}| = \frac{|\boldsymbol{v}_1|\cos(\beta_3)(j_{Tx}g_2 + j_{Ty}g_1) - |\boldsymbol{v}_1|\cos(\alpha_1)(i_{Tx}g_2 + i_{Ty}g_1)}{h_2g_1 - h_1g_2} \tag{7.19}$$

The expression for $\boldsymbol{r}$ in terms of components of frame G is obtained by substituting (7.18) and (7.19) into (7.13) yielding:

$$\boldsymbol{r} = \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \begin{bmatrix} |\boldsymbol{v}_1|\cos(\beta_3)j_{Tx} + |\boldsymbol{v}_1|h_1 \\ |\boldsymbol{v}_1|\cos(\beta_3)j_{Ty} - |\boldsymbol{v}_1|h_2 \\ |\boldsymbol{v}_1|\cos(\beta_3)j_{Tz} + |\boldsymbol{v}_1|h_3 \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} \tag{7.20}$$

Given $\boldsymbol{r}$, we may define v_{1z} as:

$$v_{1z} = \sqrt{|\boldsymbol{v}_1|^2 - |\boldsymbol{r}|^2} \tag{7.21}$$

Then, the coordinates of the center of the camera lens are:

$$\boldsymbol{P} = \begin{bmatrix} x_1 + r_x + k_{Tx}v_{1z} \\ y_1 + r_y + k_{Ty}v_{1z} \\ z_1 + r_z + k_{Tz}v_{1z} \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} \tag{7.22}$$

We may determine the orientation of frame C relative to frame G by expressing $\widehat{\boldsymbol{i}}_c$, $\widehat{\boldsymbol{j}}_c$, $\widehat{\boldsymbol{k}}_c$ in terms of $\widehat{\boldsymbol{i}}_G$, $\widehat{\boldsymbol{j}}_G$, $\widehat{\boldsymbol{k}}_G$.

Using the interior projection angles, ϕ_i, easily determined from the known coordinates x_{ci}, y_{ci}, z_{ci}, $\widehat{\boldsymbol{k}}_c$ may be calculated as:

$$\widehat{\boldsymbol{k}}_c = \begin{bmatrix} k_{cx} \\ k_{cy} \\ k_{cz} \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \begin{bmatrix} \boldsymbol{v}_1^T \\ \boldsymbol{v}_2^T \\ \boldsymbol{v}_3^T \end{bmatrix}^{-1} \begin{bmatrix} |\boldsymbol{v}_1| \cos(\phi_1) \\ |\boldsymbol{v}_2| \cos(\phi_2) \\ |\boldsymbol{v}_3| \cos(\phi_3) \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} \tag{7.23}$$

Recognizing that $\boldsymbol{v}_4$ and $\boldsymbol{v}_5$ may be expressed in terms of frame G, as well as frame C, the following equalities may be established:

$$\boldsymbol{v}_4 = -\frac{|\boldsymbol{v}_4|}{|\boldsymbol{v}_1|}\boldsymbol{v}_1 \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \begin{bmatrix} x_{c1} \\ y_{c1} \\ -d_z \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_c \\ \widehat{\boldsymbol{j}}_c \\ \widehat{\boldsymbol{k}}_c \end{bmatrix} \tag{7.24}$$

$$\boldsymbol{v}_5 = -\frac{|\boldsymbol{v}_5|}{|\boldsymbol{v}_2|}\boldsymbol{v}_2 \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} = \begin{bmatrix} x_{c2} \\ y_{c2} \\ -d_z \end{bmatrix} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_c \\ \widehat{\boldsymbol{j}}_c \\ \widehat{\boldsymbol{k}}_c \end{bmatrix} \tag{7.25}$$

Solving (7.24) for $\widehat{\boldsymbol{i}}_c$, and substituting the result into (7.25) yields the desired expression for $\widehat{\boldsymbol{j}}_c$:

$$\widehat{\boldsymbol{j}}_c = \frac{1}{y_{c2}x_{c1} - y_{c1}x_{c2}} \left\{ x_{c1} \begin{bmatrix} x_5 \\ y_5 \\ z_5 \end{bmatrix} - x_{c2} \begin{bmatrix} x_4 \\ y_4 \\ z_4 \end{bmatrix} + (x_{c1} - x_{c2}) \begin{bmatrix} d_z k_{cx} - P_x \\ d_z k_{cy} - P_y \\ d_z k_{cz} - P_z \end{bmatrix} \right\} \bullet \begin{bmatrix} \widehat{\boldsymbol{i}}_G \\ \widehat{\boldsymbol{j}}_G \\ \widehat{\boldsymbol{k}}_G \end{bmatrix} \tag{7.26}$$

The unit vector, $\widehat{\boldsymbol{i}}_c$, can be obtained from the cross product of $\widehat{\boldsymbol{j}}_c$ and $\widehat{\boldsymbol{k}}_c$.

Finally, the orientation of frame C relative to frame G can be expressed by a rotation matrix:

$${}^G_c R = \begin{bmatrix} \widehat{\boldsymbol{i}}_c & \widehat{\boldsymbol{j}}_c & \widehat{\boldsymbol{k}}_c \end{bmatrix} = \begin{bmatrix} i_{cx} & j_{cx} & k_{cx} \\ i_{cy} & j_{cy} & k_{cy} \\ i_{cz} & j_{cz} & k_{cz} \end{bmatrix} \tag{7.27}$$

7.2.3 Image Distortion and Correction

The mathematical relationships established in Sections 7.2.1 and 7.2.2 provide a means for specifying 3D workspace coordinates from a 2D image, and for precisely determining the

position and orientation of a camera. The relationships were derived using assumptions that the camera lens and CCD are ideal elements, and therefore project perfect images of the workspace onto the touchscreen monitor. In reality, the image displayed on the monitor may be a noticeably distorted representation of the actual workspace. This is demonstrated in Figure 7.7 where the image of a square grid pattern was acquired using an Axis 210 machine vision system with a 4-millimeter focal-length lens.

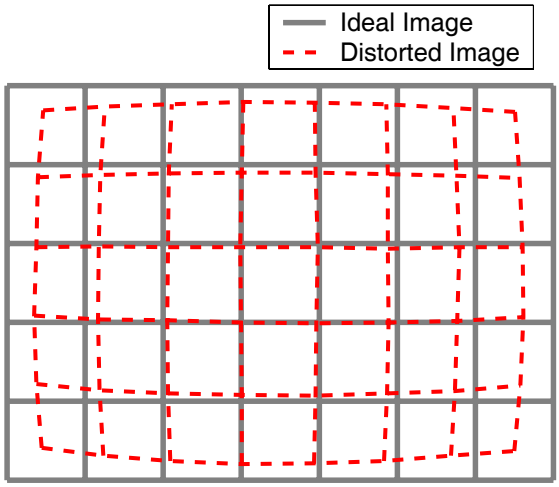


Figure 7.7: Distorted and Ideal Image of Square Grid.

The type of significant image distortion shown in this figure is characteristic of many commercially available industrial cameras. Distortions of this magnitude can adversely affect position and orientation measurements obtained using the methods of Sections 7.2.1 and 7.2.2. The precision with which an operator selects pixels on the touchscreen monitor can also contribute to the final accuracy of the calculations. Consequently, the objective of this section is to identify the elements that contribute to inaccurate calculations and present a correction technique that mitigates the most significant sources of these errors.

Understanding the process of how an image is acquired by a camera, and afterward used by an operator, is an important step in identifying sources of error. Figure 7.8 illustrates this process. Light rays reflecting off an object pass though a camera lens. The rays are redirected toward a CCD, and comprise an image of the object. Photoelectric cells on the CCD digitize the image, which is subsequently encoded and electronically sent to

the touchscreen. The digitized image is uncompressed and replicated on the pixels of the monitor. Finally, the operator touches the screen, selecting the image of the desired location.

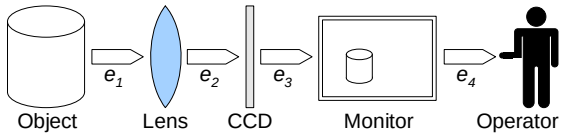


Figure 7.8: Image Acquisition and Selection Process.

In this process, four sources of error (denoted by e_{1-4} in the figure) can contribute to inaccurate position and orientation calculations. Error e_1 represents the distortions of the light incident upon the camera lens. These distortions can be caused by variations in air density (such as those seen rising from asphalt on a hot day) or other changes in the surrounding medium's index of refraction. The error e_2 is the result of two physical aspects of the camera's construction: lens shape and CCD alignment. Surface imperfections on the lens cause incident light rays to be misdirected toward the CCD. Misalignment of the CCD alters the focal length of the camera. Error e_3 is caused by the loss of data generally occurring when the digitized image is compressed with an encoding scheme. Finally, e_4 is due to the pixilation of the image as it is displayed on the monitor. This affects the ability of the operator to precisely select the pixel of the desired location.

In many circumstances, the error e_2 contributes most significantly toward incorrect position and orientation calculations by drastically distorting images captured by the CCD. The lens surface imperfections and CCD misalignment constituting e_2 cause translational and spherical aberration abnormalities in the images. Accordingly, these distortions can be corrected through a coordinate transformation that maps the coordinates of distorted pixels to coordinates of pixels that would comprise an undistorted image. The corrected coordinates may be found from:

$$\begin{bmatrix} x_{ci} \\ y_{ci} \end{bmatrix} = s_i \begin{bmatrix} x_{di} + \Delta_x \\ y_{di} + \Delta_y \end{bmatrix} \tag{7.28}$$

where $(x_c,\, y_c)_i$ are the corrected coordinates, $(x_d,\, y_d)_i$ are the distorted coordinates, $(\Delta_x,\, \Delta_y)$ are the translational correction terms, and s_i is the spherical aberration correction

term. Whereas the translational correction terms remain constant for each pixel coordinate pair, the value of s_i depends on the coordinate pair being transformed. This term can be derived from a polynomial series expansion about the translated coordinate pairs:

$$s_i = 1 + \sum_{k=1}^{n} C_k \Big[(x_{di} + \Delta_x)^2 + (y_{di} + \Delta_y)^2 \Big]^{k/2} \tag{7.29}$$

In practice, the weighting coefficients, C_k, converge rapidly to zero. As a result, only the first few terms in the series are needed. Determining the values of the weighting coefficients and the translational correcting terms is easily accomplished through a numerical optimization routine that minimizes the error between the transformed coordinate pairs and the coordinate pairs associated with an undistorted image.

This procedure was performed in order to correct image distortion caused by the Axis 210 camera. Figure 7.9 displays the results of the correction together with an undistorted image for comparison. The corrected image aligns closely with the undistorted image. These results indicate that the correction scheme enables non-ideal cameras to perform nearly ideally.

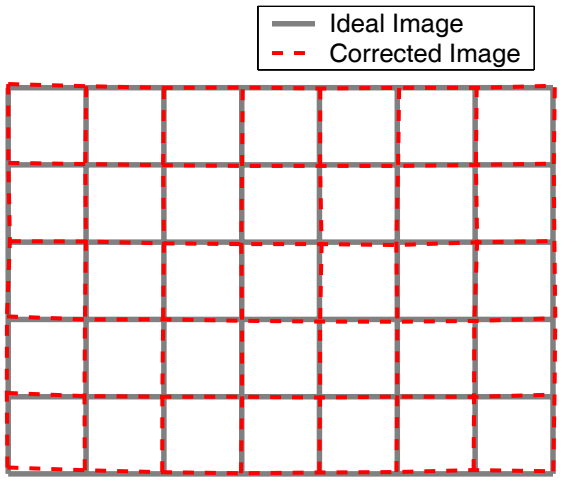


Figure 7.9: Corrected and Ideal Image of Square Grid.

Accordingly, implementing the distortion correction scheme will enable the calculation methods of Sections 7.2.1 and 7.2.2 to be utilized with non-ideal cameras exhibiting significant image distortion due to misalignment and spherical aberration.

7.2.4 Interface Precision Capabilities

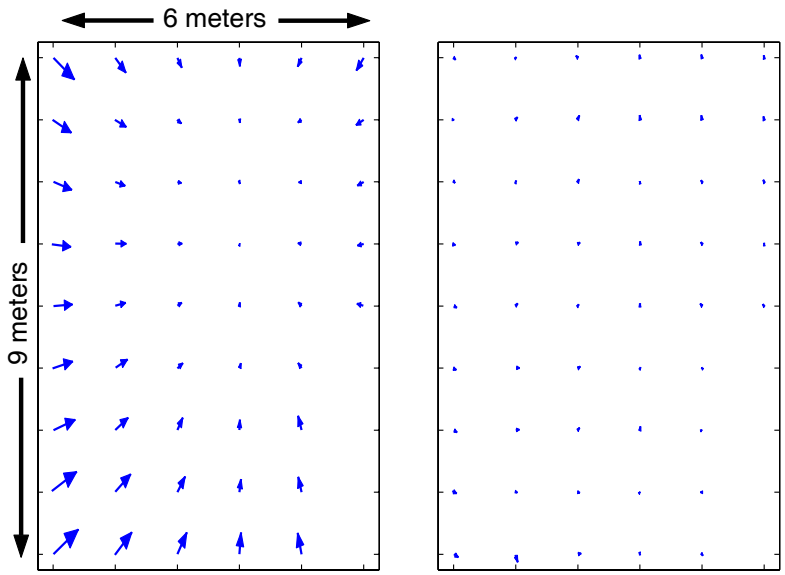
The Axis camera was positioned to look downward onto the workspace of the 10-ton crane described previously in Section 3.1.4. To evaluate the accuracy of the visual interface in determining workspace coordinates, first the camera was calibrated using the method of Section 7.2.2. Then, the coordinates of 50 points in the workspace were calculated using the methods presented in Sections 7.2.1 and 7.2.2. The test points used for the experiment were obtained from the vertices of a 6 m by 9 m grid that was drawn on the workspace floor. The grid consisted of 1 m by 1 m partitions. While viewing the image from the touchscreen monitor, the operator selected each of the test points by carefully touching them with a pointer. The workspace coordinates of the test points were then calculated from the coordinates of the touched pixels. The errors of these calculations were determined by comparing the calculated positions to the known positions of the 50 test points. This process was completed both with and without the distortion correction technique presented in Section 7.2.3.

The results of this experiment are represented graphically in Figure 7.10. Figure 7.10(a) displays the error of the calculated measurements with the distortion correction algorithm disabled. The tail of each arrow originates at the actual position of the respective test point. The arrows terminate at the calculated positions of each test point. One observes that the points farthest from the image center have the largest measurement error. This result was anticipated because of the spherical aberration distortion caused by the lens contour. The average error for the 50 points was 21.5 cm.

With the distortion correction algorithm enabled, the position calculation errors were significantly reduced. These results are shown in Figure 7.10(b). The average calculation error was reduced to 4.5 cm.

7.3 Conclusions

Human operation of cranes involves both supervisory tasks coupled with manipulation tasks. Models of human/crane interaction were developed and used to identify aspects of these



(a) Uncorrected. (b) Corrected.

Figure 7.10: Position Calculation Errors for Uncorrected and Corrected Lens Distortion.

tasks where technological changes could simplify crane operation. This investigation motivated the development of a visual touchscreen interface. The interface permits operators to command crane motion by pointing at an image of a desired destination. This method of crane manipulation is simpler for the operator than conventional pendent or lever-type control. Mathematical relationships were derived that allow coordinates of an image selected by an operator to be converted into coordinates of a desired workspace position. The transformation is effective even when images exhibit distortion due to spherical aberration and pixel translation. Experimental results demonstrated that workspace coordinates could be specified to within 5 *cm* of accuracy.

CHAPTER VIII

OPERATIONAL EFFECTS OF CRANE INTERFACE DEVICES[1]

CHAPTER SUMMARY: *The principal difficulty associated with crane manipulation is payload swing. Consequently, dynamic compensation that suppresses this behavior, such as input shaping, can significantly improve operational efficiency. However, operational efficiency is also influenced by the human/machine interface. This chapter examines additional benefits that may arise from changing the way operators interact with dynamically compensated cranes. The results of several operator studies are presented to demonstrate these advantages. This chapter also presents a crane trajectory classification system. The utility of this construct is that it provides a means of identifying specific attributes of manipulation scenarios that can influence operational efficiency. In this way, the classification system provides a framework with which to evaluate results from the operator studies.*

The flexibility of cranes leads to undesirable payload oscillation. This makes it challenging for human operators to manipulate the structures. Input shaping reduces payload oscillation, and can, therefore, improve operational efficiency. Yet, efficiency can be further improved by changing the way operators interact with cranes.

Chapter 7 provided a basic understanding of how the human/crane interface affects crane operators. The models of human/crane interaction presented in Chapter 7 motivated the development of the visual touchscreen interface, which seeks to simplify the role of the human operator. The type of gesture-based control enabled by the visual interface utilizes intuition-based behavior [3, 21], which is less complex and requires less cognitive resource use than manual pendent or joystick control. A tremendous benefit of "shifting" operator actions toward intuition-based behavior is that more mental resources are free to conduct

[1] Portions of this chapter are published in [110, 89, 109].

supervisory tasks, such as monitoring, assessing potential hazards, and payload transport decision making [113].

While the work presented in Chapter 7 provided a theoretical foundation for anticipating advantageous benefits from using the visual interface, experimental validation of these benefits were not presented. Consequently, the objective of this chapter is to:

- *Assess the operational effects of pendent, joystick, and visual interface devices on cranes which use dynamic compensation.*

To assist in completing this objective, Section 8.1 describes a crane motion classification system, which provides a framework for evaluating the influence of different interface devices on operational efficiency. The remainder of the chapter describes crane operator experiments that were conducted to test the efficiency effects of different interfaces over several classes of crane trajectories. Data gathered from these experiments, in conjunction with the classification system, is used to draw conclusions about the operational effects of different interface devices.

8.1 Classification System and Performance Measures

The utility of the classification system presented here is that it provides a means with which important crane trajectory characteristics can be well defined. By clearly defining trajectory characteristics, distinctions between different types of crane motions can be more easily identified. This capability can assist researchers in evaluating the effects of different interface devices, especially in the context of crane operator studies. For example, suppose that interface A is used to manipulate a crane over a given trajectory. Then, interface B is used to manipulate the crane over the same trajectory, which results in a faster completion time. Given this result, one may quickly ascribe interface B with superior attributes. However, without an objectifying perspective, one cannot be certain whether or not this result can be attributed to superior attributes of interface B, or inherent properties of the tested path, which, in this instance, were unfavorable to interface A. The classification system attempts to provide an objectifying framework that can help evaluate interface performance in a general sense, and not merely on a case-by-case basis.

Section 8.1.1 discusses *path-based classifications*. These classes describe the properties that are intrinsic to a given payload trajectory. Section 8.1.2 presents *path and interface-based classifications*. These categories describe attributes that depend on both the payload trajectory and the interface device being used by an operator.

8.1.1 Path-Based Classifications

Path-based classifications seek to identify attributes of a given crane trajectory that are explicitly associated with the path and its workspace.

8.1.1.1 Nominal Trajectory

This classification captures the "directness" of a payload path from an initial starting position to a final position. The *nominal trajectory* classification is specified as either **Direct** or **Indirect**. A path classified as a direct nominal trajectory is defined as the trajectory requiring the shortest time duration to travel.

Figure 8.1 illustrates the difference between these two trajectories. This figure shows an overhead view of a hypothetical hoisting scenario. Objects A, B, and C are to be maneuvered onto a truck while avoiding obstacles. Various possible paths are shown. Object B could be maneuvered to region $\bar{B}$ *directly* along path 1 or *indirectly* along path 2. Therefore, the nominal trajectory classification for path 1 and path 2 is direct and indirect, respectively.

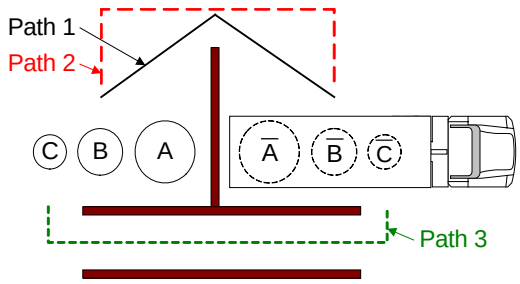


Figure 8.1: Overhead View of a Hoisting Scenario.

8.1.1.2 Path Deviation

Maneuvering a crane through a cluttered work environment presents an obstacle avoidance problem for an operator. The presence of obstacles constrains the amount by which an operator can deviate from a given path without causing a collision. Obviously, different payload paths have varying degrees of allowable deviation. For example, while maneuvering a payload along path 2 of Figure 8.1, an operator could significantly deviate from the path and not cause a collision with an obstacle. On the other hand, deviating from path 3 could more easily result in a collision.

Path deviation in the horizontal plane of a payload is generally more difficult to control than vertical path deviation. This is attributable to the fact that operators can directly control vertical payload position through hoisting. In contrast, the horizontal payload position depends on both the position of the overhead support, which operators can directly control, as well as cable-sway, which operators have difficulty controlling.

To quantify the horizontal maneuverability difficulty associated with navigating through a cluttered work environment, a *path deviation* parameter, D, is defined:

$$D = \frac{\frac{1}{2}\oint \left(h_1(s) + h_2(s)\right) ds}{\oint ds} \tag{8.1}$$

The symbol, s, is the arc length along a given path. $h_{1,2}(s)$ are the permissable path deviation widths at each point along the path. $h_1(s)$ represents the permissable deviation width on one side of the path, while $h_2(s)$ represents the permissable deviation width on the other side. The path deviation parameter, D, is physically meaningful. It represents the average permissable deviation of a payload to either side of a path, calculated over the entire length of the path. Thus, for highly constrained paths, the value of D is small, and for less restrictive paths, the value of D is large.

In some manipulation scenarios, there are segments of a given path over which there are no obstacles to avoid. According to the strict definition of $h_{1,2}(s)$, these instances result in $h_{1,2}(s)$ being set equal to infinity. Consequently, the value of D also approaches an infinite value. As a result, other portions of the given path, which may require obstacle avoidance, are not adequately represented in the path deviation parameter. In order to permit D to

more accurately represent these scenarios, the value of the deviation parameters, $h_{1,2}(s)$, are limited to a suitably large value: $h_{max} = 3A$. The meaning of the parameter, A, and significance of setting h_{max} equal to $3A$ is discussed shortly.

The value of the deviation parameter, D, can be normalized to the size and capability of the crane to which it is being applied. This is accomplished by measuring the distances, $h_{1,2}(s)$, in units of another distance, A, defined as:

$$A = 2V_m\sqrt{L/g} \tag{8.2}$$

where V_m, L, and g are the maximum velocity of the crane, suspension cable length, and acceleration due to gravity, respectively. The distance, A, is equal to the peak-to-peak swing amplitude exhibited by a crane when it is quickly accelerated from rest to full velocity. An illustration of this nominal swing distance is shown in Figure 8.2. Because of the explicit dependence of A on the crane configuration (L), and velocity capability (V_m), the value of A provides a physically meaningful and easily calculated normalizing factor.

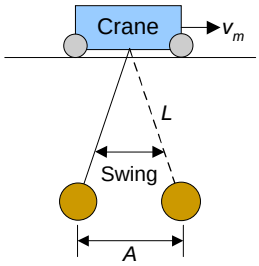


Figure 8.2: Oscillating Load Caused by Crane Acceleration.

In light of its physical significance, one can more easily understand why h_{max} is set equal to a multiple of A. Because A represents nominal swing amplitude, the distance $h_{max} = 3A$ represents a very wide permissable path deviation width in relation to anticipated hook swing. Generally, it is simple, even for an inexperienced crane operator, to manipulate a payload within a region defined by $h_1(s) = h_2(s) = 3A$, yet, this distance is not so large as to cause resulting values of the path deviation parameter, D, to misrepresent the difficulty of maneuvering a payload through both cluttered and uncluttered portions of a given trajectory.

As a simple example of how the deviation parameter can be used to capture the difficulty

of obstacle avoidance during a maneuvering task, consider, Figure 8.3(a), which depicts a simple 6 m, straight-line, path. Two regions of this path are unencumbered by obstacles. Accordingly, the permissable path deviation over these regions is $3A$. Thus, for $0 \leq s \leq 4\ m$ and for $5\ m \leq s \leq 6\ m$, both $h_1(s)$ and $h_2(s)$ equal $3A$. One portion of the path is restricted by an obstacle of width 1 m. In this portion of the path, the amount by which an operator can deviate from one side of the path without hitting the obstacle is $0.5A$. Thus, for $4\ m \leq s \leq 5\ m$, $h_1(s)$ may be set equal to $0.5A$, which implies that $h_2(s)$ is equal to $3A$. Given these values of $h_{1,2}(s)$ over the length of the path, the deviation parameter, D, can be calculated as:

$$D = \frac{\frac{1}{2}\Big[(3A+3A)4\ m + (0.5A+3A)1\ m + (3A+3A)1\ m\Big]}{6\ m} \approx 2.8A$$

Using a similar procedure, the deviation parameter value for the path shown in Figure 8.3(b) is calculated:

$$D = \frac{\frac{1}{2}\Big[3(0.5A+3A)1\ m + 3(3A+3A)1\ m\Big]}{6\ m} \approx 2.4A$$

Thus, the small and large values of D obtained for the two considered paths indicate that the first path is less restricted than the second path. In the first scenario, the average distance that a payload could deviate to either side of the desired path is $D = 2.8A$, whereas in the second scenario, this average distance is reduced.

(a) Path with Small Path Deviation Limit; $D = 2.8A$.

(b) Path with Moderate Path Deviation Limit; $D = 2.4A$.

Figure 8.3: Allowable Deviation of Two Similar Paths.

While (8.1) provides the means for calculating *exact* values of D, it should be noted that exact values of D are not necessary for obtaining a meaningful sense of average path restriction. Instead, when calculating D, reasonable simplifications should be used. This helps to facilitate rapid calculation of D, and also helps to avoid confusion that may arise from unusual trajectory/obstacle configurations.

8.1.1.3 Target Size

The target size classification reflects constraints on the final placement precision of payloads. For example, in Figure 8.1, a high level of precision is required to place object C within region $\bar{C}$. But, less precision is needed to place object C within region $\bar{A}$.

The precision with which an object is to be placed can be enumerated by a single *target size* parameter, T, that represents permissable positioning error. Because the difficulty associated with manual positioning is influenced by payload swing, the target size parameter is measured in units of the previously defined nominal swing distance, A. By so doing, the value of the target size parameter is normalized to the size and capability of the crane to which it is being applied. $T = 2$ indicates low required precision because the final placement tolerance is two times the nominal swing distance, A; $T = 0.5$ indicates a higher precision requirement because the final placement tolerance is half the nominal swing distance, A.

8.1.1.4 Collision Penalty

Cranes are used in a variety of different environments. Crane operators at the Hanford Site in Washington State, for example, routinely stack radiological packages in tight matrix formations. Because of the hazardous payload content, operators are extremely careful to avoid obstacles. On the other hand, operators in some steel mills, requiring only gross motion of scrap steel, are not overly concerned about obstacle avoidance.

The type of collision concern variance exemplified in these scenarios is accounted for by the *collision penalty* classification. The collision penalty is qualitatively defined as **Low**, **Medium**, or **High**.

A low collision penalty is assigned to a task if the consequence of a collision is, at worst, a minor inconvenience. A medium collision penalty is assigned if the consequence of a collision may result in damage to equipment. A high collision penalty results if colliding with obstacles may result in physical harm to the operator or others.

8.1.1.5 Efficiency Penalty

The *efficiency penalty* classification addresses the efficiency concern of crane operators. Operators at a high throughput shipping port, for example, are more concerned about continual movement efficiency than operators hoisting only a few payloads per day. This classification is qualitatively defined as **Low**, **Medium**, or **High**.

A low efficiency penalty is assigned to a task if there is little to no consequence for inefficient manipulation. A medium efficiency penalty is assigned if there is moderate demand for efficient motion. A limited number of deviations from an optimal trajectory are acceptable. A high efficiency penalty is assigned when a given task requires high throughput. Large throughput shipping ports exemplify scenarios with a high efficiency penalty.

8.1.2 Path and Interface-Based Classifications

Path and interface-based classifications describe path attributes that depend on both the payload trajectory and the interface device being used by an operator.

8.1.2.1 Actuation Realizability

The difficulty associated with maneuvering a crane along a given trajectory depends on both the complexity of the path and the interface device being used. For example, an operator using a push-button pendent interface is limited to maneuvering a crane in discrete directions and velocities that correspond to the various pendent button states. A joystick, in contrast, allows a crane to be manipulated in a more variable manner. Using the joystick, an operator can drive the crane in any horizontal direction with infinitely variable speed. Likewise, because the visual interface is used in conjunction with a computerized controller, it also permits the crane to drive in any direction with infinitely variable speed.

The *actuation realizability* classification captures the capacity of an interface device to allow an operator to follow a given trajectory. For a given trajectory and interface device, the actuation realizability is categorized as either **simple**, **moderate**, **complex**, or **not possible**.

- **Simple**: The nominal path is achievable using exactly 1 mode of actuation at a given

instant. For the pendent, a simple trajectory requires depressing only one button at a time. For the joystick, a simple trajectory involves any motion of the payload in the horizontal plane. For the visual interface, all achievable trajectories are classified as simple because all motion is accomplished by the controlling computer.

- **Moderate**: The nominal path is achievable using 2 modes of actuation at a given instant. This corresponds to using two buttons simultaneously on the pendent, or using the joystick to move the payload horizontally, while also hoisting the load with an "up-down" button. All motions made by the joystick are classified as either simple or moderate.

- **Complex**: The nominal path is achievable using 3 or more modes of actuation at a given instant. Only the motions made using the pendent can be classified as complex.

- **Not Possible**: The nominal path is not achievable using any combination of actuation modes. For the pendent, a path classified as not possible could be at some obscure angle that does not correspond to any combination of depressed buttons.

8.1.2.2 Viewing Perspective

The *viewing perspective* that an operator has while manipulating a crane can affect operation difficulty. An operator's perspective can be localized in the crane workspace, and either near to the object being hoisted or far from it. Likewise, an operator manipulating a payload remotely, through a video surveillance or other remote capacity, may have a close-up or zoomed-out view. In some instances an operator may have a mixed perspective where local viewing is supplemented by video surveillance.

A local perspective facilitates better spacial awareness by the crane operator because he or she can actively sense, and is engaged in, the 3-dimensional workspace. Remote crane operation is generally more difficult because 3-dimensional information must be extrapolated from 2-dimensional images. In either case, local or remote operation, precision positioning tasks are more easily accomplished when the operator has a close-up view of the payload and target region.

The viewing perspective classifications capture the various viewing configurations. These classifications are **local-near**, **local-far**, **remote-near**, **remote-far**, **mixture-near**, and **mixture-far**.

8.2 Crane Operator Studies

Several operator studies were conducted in order to determine the operational effects of different crane interfaces on systems using dynamic compensation. The test described in Section 8.2.1 involves a conventional interface that is used to manipulate a crane both with and without input shaping. The results from this series of experiments demonstrate that significant efficiency benefits result from the use of dynamic compensation. In order to determine if additional performance benefits may be derived by changing the operator interface, Sections 8.2.2 and 8.2.3 describe tests conduced wherein conventional interfaces, and the newly developed visual interface, are used with a dynamically compensated crane.

8.2.1 Influence of Input Shaping on Operational Performance

To determine the beneficial attributes of input shaping on conventionally controlled cranes, an operator study was conducted in a simulated work environment [55]. This section presents several key results and figures from that study.

A simulated work environment was erected in the workspace of the crane previously described in Section 3.1.4. A photograph of the constructed workspace is shown in Figure 8.4. Ten volunteer crane operators, with varying experience, were required to manipulate a large distributed payload from the starting region to the target region, while avoiding obstacles. The payload and rigging configuration caused the crane to exhibit two-mode oscillations during movement of the crane. The nominal path along which the operators manipulated the payload is shown with the dotted line.

During a period of two weeks, each operator completed the manipulation task several times with and without a two-mode input shaper enabled. The input shaper was designed to suppress oscillation at the two frequencies that occur during uncontrolled motion. To operate the crane, operators used a conventional pendent interface, shown in the subplot of Figure 8.4. The pendent was equipped with buttons corresponding to forward, reverse,

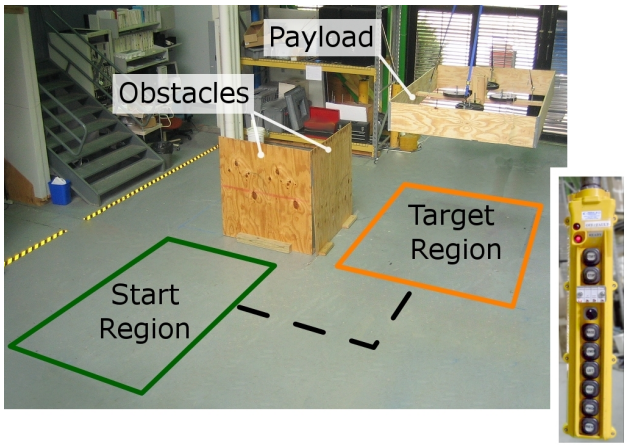


Figure 8.4: Simulated Workspace Used for Input Shaping Operator Tests.

left, right, up, and down. Diagonal motions were possible by simultaneously pressing two buttons. A fully-pressed button issued full-speed commands, while a half-pressed button issued half-speed commands. Operators were incentivized with a cash prize to complete the manipulation task quickly. The operator with the fastest completion time earned $100. Operators were also incentivized to avoid colliding with the obstacles. Each collision resulted in a five-second penalty, which was added to the total completion time.

The nominal path along which operators manipulated the payload was aligned with the crane's actuation directions. In this way, operators needed only press a single pendent button at any instant in order to traverse the path. Accordingly, the actuation realizability for this task is classified as simple. While traversing the path with the large payload, operators had to narrowly avoid obstacles. The location of the obstacles in relation to the path and size of the payload corresponds to a permissable path deviation of $D \approx 1.6A$. Both the width and length of the target region was approximately 20 *cm* larger than the footprint of the payload. For the given crane configuration, and maximum crane speed, the payload footprint and target region sizes correspond to a target size classification of $T \approx 0.35A$. This represents a moderately stringent requirement on positioning precision because the permissable positioning error was equivalent in length to approximately one-third of the vibration amplitude induced by a step command. The actuation realizability, path deviation, and target size classifications, along with other pertinent classifications for

this experimental setup, are summarized in Table 8.1.

Table 8.1: Classifications for the Input-Shaping Test Course.

Path Deviation, D	Target Size, T	Nominal Swing, A	Collision Penalty	Efficiency Penalty	Actuation Realizability	Perspective
$\approx 1.6A$	$\approx 0.35A$	$\approx 0.57m$	Medium	Medium	Simple	Local-Near

To determine how performance is affected by dynamic compensation, the time required by each operator to complete a given maneuvering task was measured. The results of these measurements are summarized in Figure 8.5, where the average completion time of every operator is shown as a function of the number of trials completed. The average completion times for the trials without input shaping are shown with the circles, the times for the trials with input shaping are shown with the squares. The error bars represent one standard deviation above and below the mean.

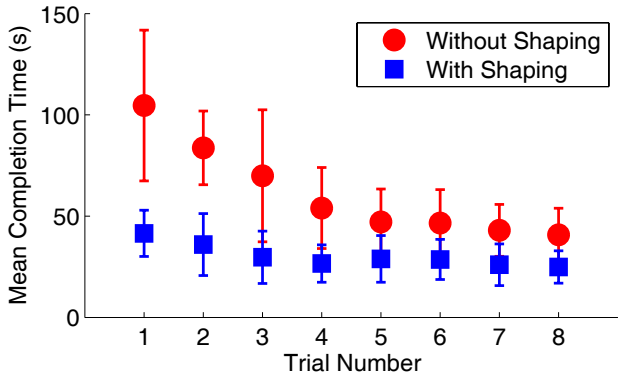


Figure 8.5: Mean Completion Times for Input Shaping Operator Tests.

Two attributes of Figure 8.5 are noteworthy. First, input shaping produced a significant reduction in average completion time over unshaped motion. Second, for both shaped and unshaped motion, there is a discernable "learning period" evidenced in the downward trend of the completion times as the number of trials is increased. These learning curves tend to flatten out for higher numbered trials. The significance of these results is that they demonstrate the beneficial attributes of effective dynamic compensation, both in terms of operational efficiency, and operator learning. With regard to operational efficiency, completion time is advantageously affected by the dynamic compensation. With regard to operator learning, dynamic compensation results in highly efficient crane operation for both

experienced and inexperienced operators.

Several other studies have verified the beneficial attributes of input shaping on conventionally controlled cranes. For the reader interested in additional resources that describes these studies, representative publications include [76, 40, 43, 124].

8.2.2 Performance Evaluation of the Visual Interface

The capability of the visual interface to improve crane operation was thoroughly evaluated. The objective was to determine whether or not operational performance could be improved, beyond what is achievable with dynamic compensation alone, by changing the way operators issue commands to cranes. To accomplish this, novice, intermediate, and expert operators manipulated a dynamically compensated crane using either a pendent, joystick, or touchscreen interface.

Both the joystick and visual interface are dissimilar to the pendent. The joystick permits the crane to be driven in any direction at an infinitely variable speed. Likewise, with the touchscreen, the crane can maneuver in any direction at an infinitely variable speed. In contrast, as described in Section 8.2.1, the pendent permits crane motion only in discrete directions and velocities that correspond to the different pendent button states. Operators were given thorough instructions pertaining to the use of the different interfaces. Additionally, participants were permitted to practice with each interface until he or she was comfortable with its use.

The dynamic compensation that was implemented during the studies is described in [108]. It is a combination of input shaping, which is used to suppress motion-induced oscillation, and feedback control, which is used to suppress disturbance-induced oscillation. The control caused the crane's payload to respond to commanded motion with minimal cable sway, regardless of which operator interface was used.

Two operator studies were conducted. For the first, participants completed a series of simple point-to-point maneuvers. The second study involved more complicated maneuvers, including hoisting of the load. Descriptions and results of these studies are presented in the following sections.

8.2.2.1 Operator Study: Simple Maneuvers

Eleven operators manipulated the crane along the three trajectories illustrated in Figure 8.6, while trying to avoid a centrally located obstacle. The three target regions comprised the starting and ending points for the three point-to-point maneuvers: AB, BC, and CA. A maneuver was not considered complete until the operator successfully stopped the crane's hook within the target region.

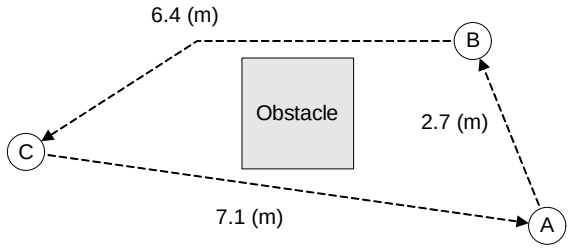


Figure 8.6: Top View of Simple-Maneuver Test Course.

Each target region had a diameter of 0.2 m; the hook diameter was approximately 0.15 m. Given the crane configuration and maximum speed capabilities, the hook and target size dimensions correspond to a target size of $T \approx 0.09A$. This value of T represents a significantly higher requirement on positioning precision than that of the previously presented operator study. The location of the obstacle in relation to the three point-to-point trajectories corresponds to a permissable path deviation of $D \approx 2.6A$. This value of D represents a less restricted workspace than that of the previous study. While participants were encouraged to complete the manipulation tasks quickly, and to avoid the obstacle, neither incentives nor penalties were awarded. The target size, path deviation, and other relevant path-based classifications for this series of trajectories is summarized in Table 8.2.

Table 8.2: Path-Based Classifications for the Simple-Maneuver Test Course.

Path Deviation, D	Target Size, T	Nominal Swing, A	Collision Penalty	Efficiency Penalty
$\approx 2.6A$	$\approx 0.09A$	$\approx 0.51m$	Low	Low

During this series of tests, the operator perspective was varied. Operators using the pendent were permitted to walk along side the hook while maneuvering the crane. However, while using the joystick or visual interface, the operators were constrained to distantly

position themselves at the edge of the workspace. While at this vantage point, the operators could only distantly see the actual hook and target regions. However, the operators could also see the workspace through a live video feed, which was also located at the edge of the workspace. The perspective classifications are based on both the path and the interface used by the operator. These classifications are summarized in Table 8.3 for the three trajectories of this study.

Table 8.3: Perspective Classification for the Simple-Maneuver Test Course.

	Pendent	Joystick	V. Interface
AB	Local-Near	Local-Far	Mixed-Far
BC	Local-Near	Local-Far	Mixed-Far
CA	Local-Near	Local-Far	Mixed-Far

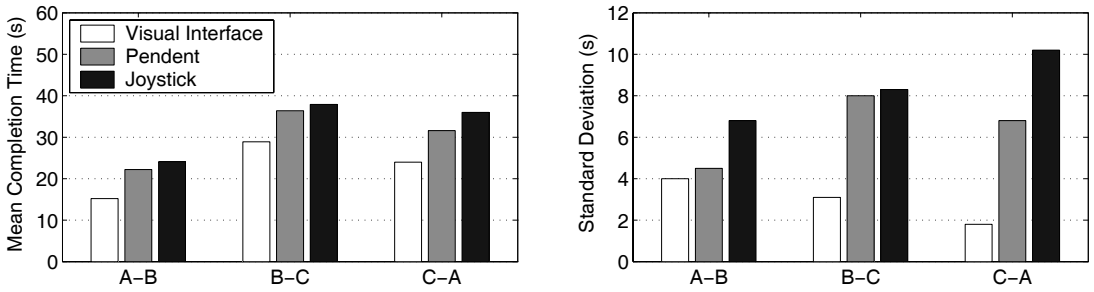
A summary of the actuation realizability for the different trajectories and interfaces is presented in Table 8.4. Note that for the case of pendent control, the actuation realizability is classified as moderate. This is because the orientation of the various trajectories are not entirely aligned with the actuation directions of the crane, therefore, two buttons needed to be pressed simultaneously in order to closely follow the nominal paths.

Table 8.4: Actuation Realizability Classification for the Simple-Maneuver Test Course.

	Pendent	Joystick	V. Interface
AB	Moderate	Simple	Simple
BC	Moderate	Simple	Simple
CA	Moderate	Simple	Simple

The times required for the operators to complete each point-to-point maneuver were recorded. These results are summarized in Figure 8.7. Figure 8.7(a) shows the average completion times of every operator for the three trajectories and respective interfaces. The visual interface reduced completion time by approximately 10 s when compared to the joystick interface, and approximately 7 s when compared to the pendent interface. Given the short duration of each of the three trajectories, the 10 s and 7 s reductions in transient time represent significant improvements in operational efficiency.

The standard deviations of the completion times are shown in the bar graph of Figure 8.7(b). The touchscreen interface resulted in a noticeably lower standard deviation for each



(a) Average Completion Times.

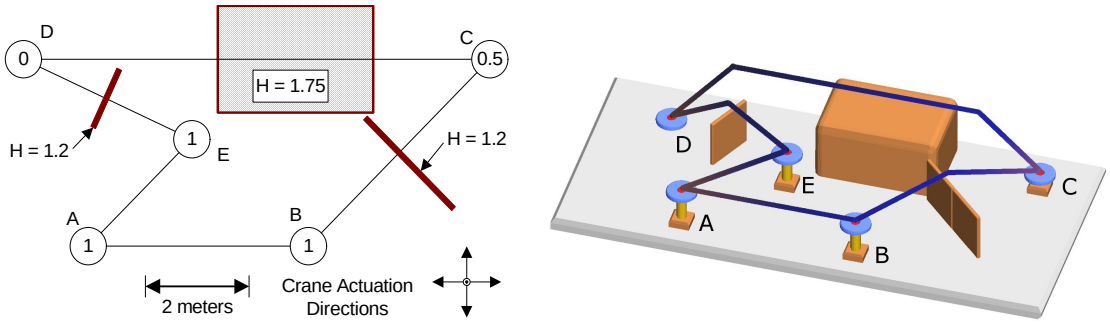
(b) Standard Deviation.

Figure 8.7: Experimental Performance Results: Simple Maneuver Test.

of the three point-to-point maneuvers. This result indicates that there was less variance in completion times among those using the visual interface than among those using either the joystick or pendent interfaces. This result is significant given that the visual interface also allowed operators to manipulate the crane more quickly than with the other interfaces.

8.2.2.2 Operator Study: Difficult Maneuvers

A test course that contained more difficult trajectories than those of the previous operator study was erected. An overhead view and 3D illustration of this test course is shown in Figure 8.8.



(a) Overhead View of Test Course.

(b) 3-Dimensional Illustration of Test Course with Nominal Paths Shown.

Figure 8.8: Illustrations of Difficult-Maneuver Test Course.

Five circular target regions comprised the starting and ending points for five point-to-point maneuvers: AB, BC, CD, DE, and EA. The height of each target region above the floor (in meters) is indicated by the number shown in each circle. The nominal hook path for each of the maneuvers is sketched with solid lines in Figure 8.8(b). Note that for three of the

five trajectories, hoisting of the load is required. As in the previous study, individual point-to-point maneuvers were not considered complete until the operator successfully stopped the crane's hook within the target region. The target size and path deviation classifications for this test course are close to those of the previous study, which had a very severe positioning requirement, and a relatively uncluttered workspace. These classifications, as well as other relevant path-based classifications, are summarized in Table 8.5 for the set of trajectories tested here.

Table 8.5: Path-Based Classifications for the Difficult-Maneuver Test Course.

Path Deviation, D	Target Size, T	Nominal Swing, A	Collision Penalty	Efficiency Penalty
$\approx 2.7A$	$\approx 0.09A$	$\approx 0.53m$	Medium	Low

The purpose of arranging the course in the configuration shown was to obtain a large range of actuation realizability among the different maneuvers while keeping other classifications relatively constant. In this way, the efficiency effects of the tested interfaces could be assessed and correlated to actuation realizability without significant influence of other varying classifications.

Path AB is the simplest trajectory. It consists of a straight-line motion without hoisting of the load. Its direction is aligned with one of the crane actuation directions. For path BC, the load must be hoisted over a barrier. The most direct path from B to C is along a 45° vertical plane. To maneuver the hook along a line in this plane, the crane must be actuated simultaneously in two directions. This is achievable with the pendent by depressing two pendent buttons at the same time. If hoisting is to be accomplished at the same time as moving along the line, a third pendent button must be depressed. Path CD is similar to path AB except it involves hoisting of the load to avoid a large obstacle. Path DE is the most difficult path among the five. The most direct path from D to E is along a non-45° vertical plane. Using the pendent, it is not possible to achieve motion in this plane. The last path, EA, does not involve hoisting. The direct path from E to A is along a 45° plane. The actuation realizability for the five *direct* point-to-point motions is summarized in Table 8.6.

Table 8.6: Actuation Realizability Classification for the Difficult-Maneuver Test Course.

	Pendent	Joystick	V. Interface
AB	Simple	Simple	Simple
BC	Complex	Moderate	Not Possible
CD	Moderate	Moderate	Not Possible
DE	Not Possible	Moderate	Not Possible
EA	Moderate	Simple	Simple

Note that for the visual interface, actuation realizability is either simple or not possible. The "not possible" classification is assigned to the visual interface when hoisting of the load is required to avoid an obstacle. As described in Chapter 7, in these scenarios the operator using the visual interface specifies the travel height of the hook in order to avoid an obstacle. In response to the operator's specifications, the controlling computer raises the hook to the specified height *before* moving the hook horizontally. After the horizontal travel is completed, and the hook is above the target region, it is lowered to the desired height. Because of this mode-of-operandi, direct motions involving hoisting while moving horizontally are not possible.

The perspective classifications for each of the trajectory and interface combinations are summarized in Table 8.7. While controlling the crane with the pendent or joystick, operators were permitted to walk along side the hook during crane maneuvers. However, because of the placement of some barriers, target regions C and D were more difficult for operators to see than the other targets. While using the visual interface, operators were confined to the edge of the workspace, but could view a live video feed of the crane and crane workspace.

Table 8.7: Perspective Classification for the Difficult-Maneuver Test Course.

	Pendent	Joystick	V. Interface
AB	Local-Near	Local-Near	Mixed-Far
BC	Local-Far	Local-Far	Mixed-Far
CD	Local-Far	Local-Far	Mixed-Far
DE	Local-Near	Local-Near	Mixed-Far
EA	Local-Near	Local-Near	Mixed-Far

Nineteen crane operators were tested on this course. Typically, operators completed

the course three or four times with each interface. Some operators completed the course as many as eight times, while others only completed the test one time.

Some typical hook trajectories for the different interface devices are shown in Figure 8.9. The joystick and pendent trajectories tend to "wander" about the region of the target locations, as well as along the nominal path. This reflects the difficulty associated with manual positioning. In contrast, the visual interface trajectories move to the intended locations without superfluous motion. Thus, for trajectories that do not involve hoisting, one would rightly expect the visual interface to produce more efficient motions.

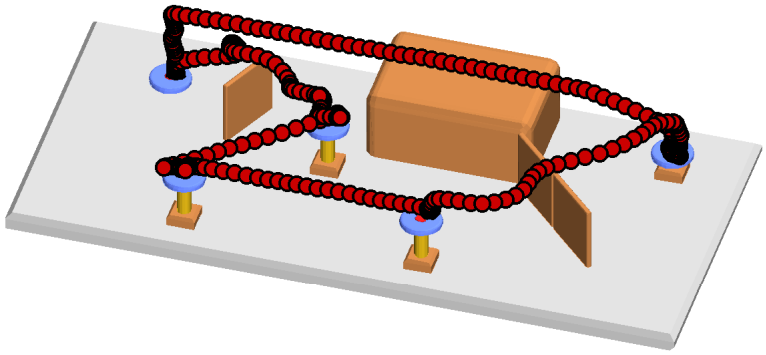
It is also evident from Figure 8.9 that the visual interface did not move the payload simultaneously in the vertical and horizontal directions during the maneuvers involving hoisting. As described previously, this is attributed to the fact that, when under the control of the visual interface, the crane raises the hook to a specified height before traveling horizontally. Hence, for trajectories involving hoisting, the visual interface does not move the hook along the most direct path. Thus, for scenarios involving hoisting, the pendent and joystick are capable of realizing more efficient motion than the visual interface.

A fundamental question arising from the apparent strengths and weaknesses of each device is: "*Is the lack of superfluous motion exhibited by the visual interface outweigh the efficiency penalties of moving along an indirect path?*"

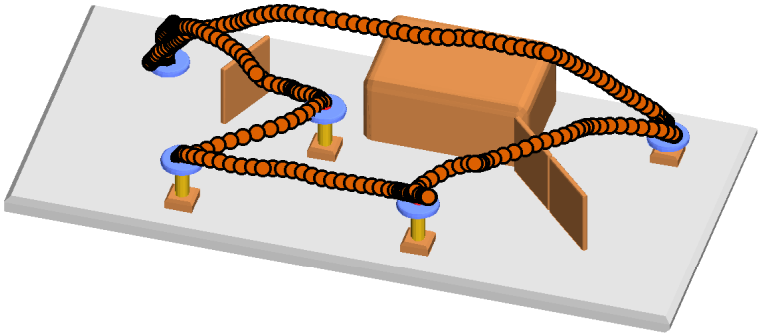
To address this question, the average completion times of every operator were determined for each path and interface. A bar graph summarizing these times is shown in Figure 8.10(a). A version of this bar graph in which the data is normalized is shown in Figure 8.10(b).

One observes from these graphs that the visual interface reduced completion time on four out of the five paths: AB, BC, DE, and EA. A 20% to 45% improvement in completion time was exhibited. For path CD, however, the average completion time for the pendent was roughly 10% faster than either the joystick or the visual interface.

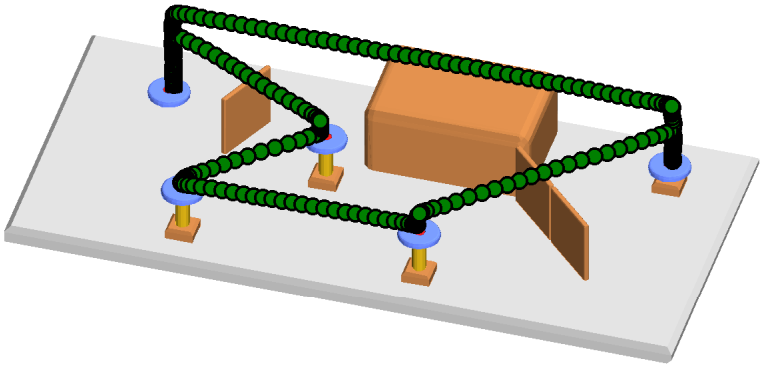
Excluding path CD, the visual interface clearly provides substantial efficiency advantages over both the pendent and the joystick. For paths not involving hoisting, this result was anticipated due to the positioning ease of the visual interface, and its lack of superfluous motion. For hoisting paths BC and DE, the results suggest that although the joystick and



(a) Pendent Trajectory.



(b) Joystick Trajectory.

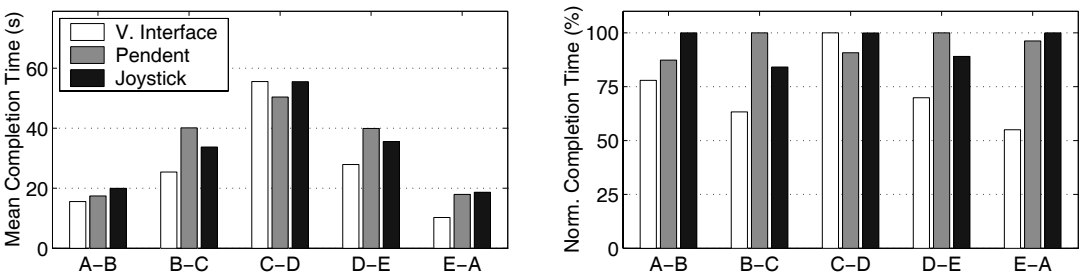


(c) Visual Interface Trajectory.

Figure 8.9: Typical Trajectories Resulting from Different Interface Use.

pendent were capable of generating more efficient trajectories than the visual interface, operators found it difficult to realize these trajectories.

The effects of operator learning also influence the efficiency of the interfaces. Learning trend plots are shown in Figure 8.11. These plots show the individual completion times for each operator as a function of how many times the operator drove through the course. The curves shown in Figure 8.11 are for path BC, but typical of all the paths. The significant



(a) Average Completion Times.

(b) Normalized Completion Times.

Figure 8.10: Experimental Performance Results: Difficult Maneuver Test.

feature of the joystick and pendent graphs is the wide variation in completion times and the notable "learning" period evidenced in the downward trend of the lines. These features are not seen on the visual interface graph. This result indicates that both novice and experienced operators using the visual interface were able to manipulate the crane in a very similar manner.

8.2.3 Operator Interface Preference

Sections 8.2.2 and 8.2.1 demonstrated that efficiency benefits, beyond those obtained through dynamic compensation, can be derived by changing the way operators drive a crane. This section considers operator preference for a given interface.

To assess the preference that operators have for a given interface, fourteen volunteers manipulated the crane through the two trajectories illustrated in Figure 8.12. Each operator completed the manipulation task three times. Once with the pendent, once with the joystick, and once with the visual interface. During each trial, the crane was enabled with the same dynamic compensation scheme as that of the previous operator studies in Section 8.2.2.

The two trajectories, AB and BC, were both narrowly constrained by obstacles. The path deviation classification for these trajectories was $D \approx 1.4$ and $D \approx 1.8$, respectively. Each trajectory also had a stringent final positioning requirement. The target size classification for both trajectories was $T \approx 0.09$. Path AB was considerably easier than path BC to navigate with the pendent. This was because path AB could be traversed by depressing two buttons simultaneously, whereas path BC required continuous trajectory adjustment

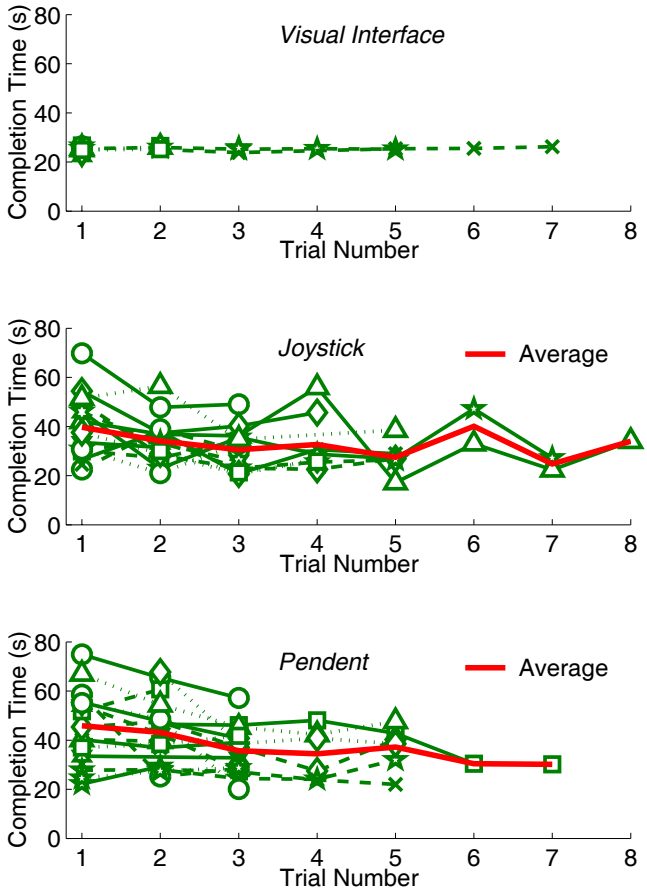


Figure 8.11: Learning Trend for Different Interface Use: Path B-C.

on the part of the operator.

Average path completion times for each trajectory and interface combination had a similar variance as those of the previous studies. The visual interface noticeably improved completion time when compared to either the joystick or the pendent. However, the primary objective of this study was to determine interface preference. To this end, after finishing the manipulation tasks, each operator completed a three-question survey:

> **Survey Question #1.** Which interface made maneuvering the crane along the point-to-point trajectories easiest?
>
> **Survey Question #2.** Which interface would you prefer to use if you worked at a shipping dock, loading and unloading cargo from ships?

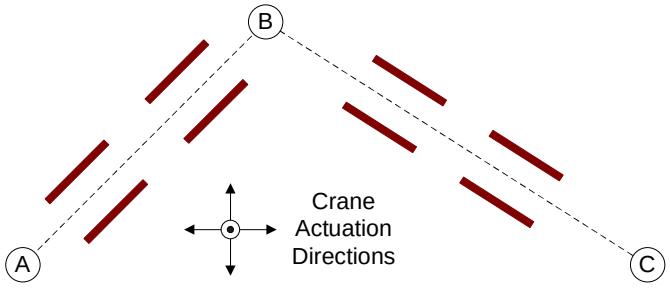


Figure 8.12: Overhead View of Desired Point-to-Point Trajectories for Interface Preference Study.

> **Survey Question #3.** Which interface provided you with the most control over the hook's trajectory?

The results from this survey are compiled in Table 8.8, where the number of responses indicating a particular interface preference are listed.

Table 8.8: Interface Preference Survey Results.

	Pendent	Joystick	V. Interface
Survey Question #1	0	2	11
Survey Question #2	0	5	9
Survey Question #3	4	7	3

The results of Table 8.8 clearly indicate a general preference for the visual interface in terms of ease-of-use. However, it is also clear that operators did not feel that the interface provided them with direct control of the crane.

In addition to the survey, operators also rated the required concentration each interface required them to exert while manipulating the crane. For this rating process, each interface could be assigned a value, 1 through 5, which correspond to the concentration level (1 = minimal, 2 = slight, 3 = moderate, 4 = high, 5 = extreme or intense). The average values assigned to each interface and path combination are summarized in Table 8.9.

Table 8.9: Required Concentration Level Survey Results.

	Pendent	Joystick	V. Interface
AB	2.7	3.2	1.1
BC	3.8	3.4	1.1

The significance of the results summarized in Table 8.9, is that they validate assertions

made in Chapter 7 regarding the cognitive process of crane operators. The models of human/crane interaction developed in Chapter 7 predicted that the visual interface would simplify crane manipulation because it is intuition-based, and removes operators from a rapidly updating control loop.

8.3 Conclusion

Efficient manipulation of cranes is enhanced by effective dynamic compensation. However, additional efficiency advantages can be obtained by changing the way operators interact with a crane. Numerous operator studies demonstrated that the visual interface facilitates faster completion times than conventional interfaces during high precision positioning tasks. The efficiency benefit was more pronounced when the desired trajectory was inherently difficult to traverse with the conventional interfaces. Furthermore, efficiency advantages were demonstrated by both novice and expert crane operators. Factors that influenced the degree to which the visual interface performed relative to the traditional interfaces were 1) actuation realizability, and 2) the difficulty operators had in physically driving the crane along a theoretically realizable path. In addition to the efficiency advantages, operators also expressed a preference for the visual interface in terms of ease-of-use. This is attributed to the intuition-based commanding procedure, and the fact that operators do not have to continually update the crane's trajectory.

CHAPTER IX

CASE STUDY: ANTI-SWAY AND POSITIONING CONTROL OF A 30-TON CRANE[1]

CHAPTER SUMMARY: *A 30-ton industrial bridge crane located at an aluminum sheet manufacturing plant has been equipped with a crane manipulation system enabling swing-free motion and precise positioning capabilities. The advancements described in previous chapters, pertaining to nonlinear input-shaping theory and interface design, together with recent developments in the areas of machine vision and feedback control, are combined into the unified crane manipulation system described here.*

Logan Aluminum is a leading manufacturer of aluminum sheet products. The manufacturing headquarters, located in Russellville, Kentucky, is equipped with modern high-speed equipment for ingot casting, hot rolling, cold rolling, and material transportation. Bridge cranes are used extensively throughout this facility for both maintenance and production tasks. The extensive crane use, coupled with the desire to increase efficiency and safety, has motivated Logan Aluminum to equip the crane fleet with advanced anti-sway and positioning capabilities. To this end, a first-of-its-kind crane manipulation system (CMS) has been developed and installed on a 30-ton crane used for maintenance purposes.

The CMS and its implementation on the Logan maintenance crane is presented in this chapter as a case study in advanced crane control. Some of the advancements presented previously in relation to nonlinear input shaping and human/crane interaction are utilized in the development of the CMS. By presenting this case study, this chapter seeks to:

[1] Portions of this chapter are published in [97, 104].

1. *Demonstrate the pragmatic application of nonlinear input-shaping theory on an industrial system.*

2. *Demonstrate how a visual control interface was implemented in an industrial setting.*

Section 9.1 describes the Logan maintenance crane, including the dynamic behavior of the system's drives and motors, and payload swing. Section 9.2 provides an overview of the CMS, and how this system is integrated into the maintenance crane. Results of performance experiments conducted on the CMS-equipped crane are presented in Section 9.3

9.1 Crane Description

Figure 9.1 shows a photograph of the 30-ton Logan crane. The trolley traverses along the bridge, which spans a distance of approximately 30 m. Likewise, the bridge can traverse along stationary runways for a distance of approximately 50 m. The hook is suspended beneath the trolley. During operation, the suspension cable length varies between 3 m and 10 m.

(a) Bridge, Hook, and Runway.

(b) Closeup of Trolley.

Figure 9.1: 30-Ton Bridge Crane at Logan Aluminum.

The bridge is equipped with two 10-horsepower, 480-volt, AC induction motors. The trolley is equipped with two 5-horsepower, 480-volt, AC induction motors. The motors are controlled by Magnetek Impulse P^3 vector drives. This equipment permits continuously variable velocity control of the structure.

Prior to installation of the CMS, operators actuated the crane by issuing commands from a lever interface, like the one shown in Figure 9.2. These commands were directly

sent to the crane drives. A model of this actuation process is illustrated with the block diagram of Figure 9.3. DM, represents the behavior of the system's vector *drives* and AC induction *motors*. This plant accepts reference velocity commands, V_r, issued to the crane by a human operator, and converts these signals to the actual velocity of the overhead trolley, V_t. Motion of the trolley causes the hook and attached payload to swing with cable angle, θ. This behavior is represented by the block, G.

Figure 9.2: Lever Crane interface. Courtesy of Magnetek.

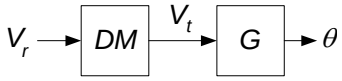


Figure 9.3: Crane Actuation Block Diagram.

9.1.1 Dynamic Analysis

The behavior of AC induction motors and vector drives is nonlinear. However, this behavior can be accurately modeled by combining several simpler components [111]. The DM block of Figure 9.3 is expanded in Figure 9.4 to reveal such a model.

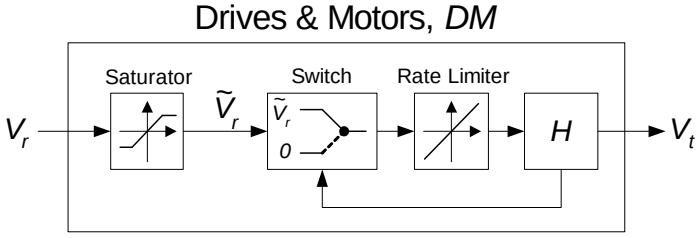


Figure 9.4: Expanded View of the Drives and Motors Model Block.

The model is comprised of four elements: a saturator, a switch, a rate limiter, and a

heavily damped second-order plant, H. The saturation element truncates excessive velocity commands to the crane, while the rate limiter places upper and lower bounds on the acceleration of the crane. H serves to mimic the smoothing behavior of the drives and motors. The function of the switching element is to pass the reference signal, $\widetilde{V}_r$, to the rate-limiting block. However, when transitional velocity commands are issued to the crane, the switch temporarily sends a signal of zero. Transitional velocity commands are those commands that change the direction of travel of the crane (forward to reverse or vise versa). This type of behavior depends on $\widetilde{V}_r$ and V_t, and can be described with the following switching rules:

$$\text{Switch Output} = \begin{cases} \widetilde{V}_r, \ \text{Sign}(\widetilde{V}_r) = \text{Sign}(V_t) \\ \widetilde{V}_r, \ |V_t| \leq X \\ 0 \ , \ \text{otherwise} \end{cases} \tag{9.1}$$

This model may be used to mimic the behavior of the maintenance crane drives and motors by properly selecting the five parameters associated with the model: p - the saturation threshold, X - the switching threshold, m - the maximum slope of the rate limiter, ζ_H - the damping ratio of H, and ω_{nH} - the natural frequency of H. For the maintenance crane, these parameters were estimated to be 0.75 m/s, 0.038 m/s, 0.63 m/s^2, 0.75, and 3.7 rad/s, respectively. Figure 9.5 compares the response of the model and the response of the actual system to several velocity commands.

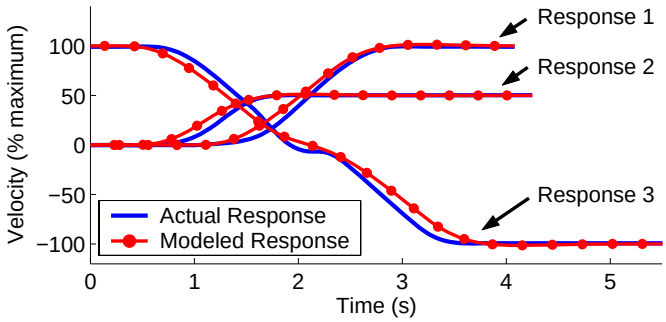


Figure 9.5: Comparison of the Actual and Modeled Responses of the Drives and Motors to Different Velocity Commands. 1) Response to a Step of 100% Velocity. 2) Response to a Step of 50% Velocity. 3) Response to a Step from 100% Velocity to -100% Velocity.

When the motors move the crane, oscillations are induced into the hook, and any attached payload. As previously illustrated in Figure 9.3, the angular response of the hook to motion of the trolley is modeled by the plant, G. This oscillatory behavior is predominantly single mode, and can be represented by a linear transfer function [108]:

$$G = \frac{\theta(s)}{V_t(s)} = \frac{(-\omega_n^2/g)s}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{9.2}$$

For the maintenance crane, the damping ratio, ζ, is approximately 0.01. The natural frequency, ω_n, is a function of the cable suspension length, L, and the acceleration due to gravity, g:

$$\omega_n = \sqrt{g/L} \tag{9.3}$$

9.2 CMS Description

The maintenance crane has been augmented with the CMS. A topological illustration of the CMS-equipped crane is shown in Figure 9.6. This figure depicts the elements that comprise the CMS:

- A control architecture for enabling swing-free motion and precise payload positioning.
- A visual human-machine interface for simplifying precise positioning tasks.
- A joystick interface for simplifying gross motion tasks.
- A standard lever interface.
- A machine vision system for sensing hook swing.
- Laser range sensors for measuring crane position.

The principal element of the CMS is the anti-sway and positioning control. This component accepts information from the other CMS elements: motion commands from the three interface devices, crane position information from the laser range sensors, and hook displacement information from the machine vision system. The information from these elements is used by the control to produce low-sway velocity commands, which are issued to the crane drives. The following subsections provide greater detail about each element of the CMS.

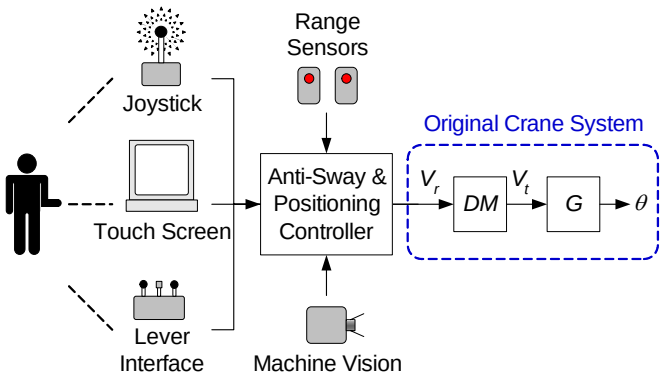


Figure 9.6: Components of the CMS Integrated into the Logan Maintenance Crane.

9.2.1 Human-Machine Interface

The three-lever interface shown previously in Figure 9.2 permits the bridge, trolley, and hook to be commanded independently from each other by their respective actuation levers. Two additional interface devices were installed with the CMS: a joystick interface, and a variation of the visual touchscreen interface described in Chapter 7. The motivation for implementing these devices was rooted in enhancing the way operators control the crane. The visual interface simplifies precise positioning tasks, while the joystick simplifies gross motion tasks.

9.2.1.1 A Visual Touchscreen Interface for Simplified Positioning

In many applications at Logan Aluminum, precise and repetitive payload positioning is required. The visual interface implemented on the maintenance crane permits operators to store desired payload destinations, and also command the crane to travel to these locations. To store a payload destination for future use, an operator first manually positions the crane at the desired location. Then, the coordinates corresponding to the crane's position can be automatically stored into the visual interface. Once the desired destination is stored, a target image that corresponds to this location is displayed on the visual interface. Operators can then command the crane to travel to the stored location by touching the image of the stored target. Once the operator specifies the desired destination in this way, the feedback control system automatically drives the crane to the specified location without payload sway.

Figure 9.7 is a screen-shot of the visual interface. In region A, the operator can store and specify hook destinations. Region B displays various system indicators, such as anti-sway activity, system errors, and operation mode. Region C displays actual and desired crane position information. This software is implemented on a wireless, hand-held, and industrially hardened touchscreen shown in the subplot of Figure 9.7. The device permits the operator to walk freely throughout the workspace in order to maintain a local perspective.

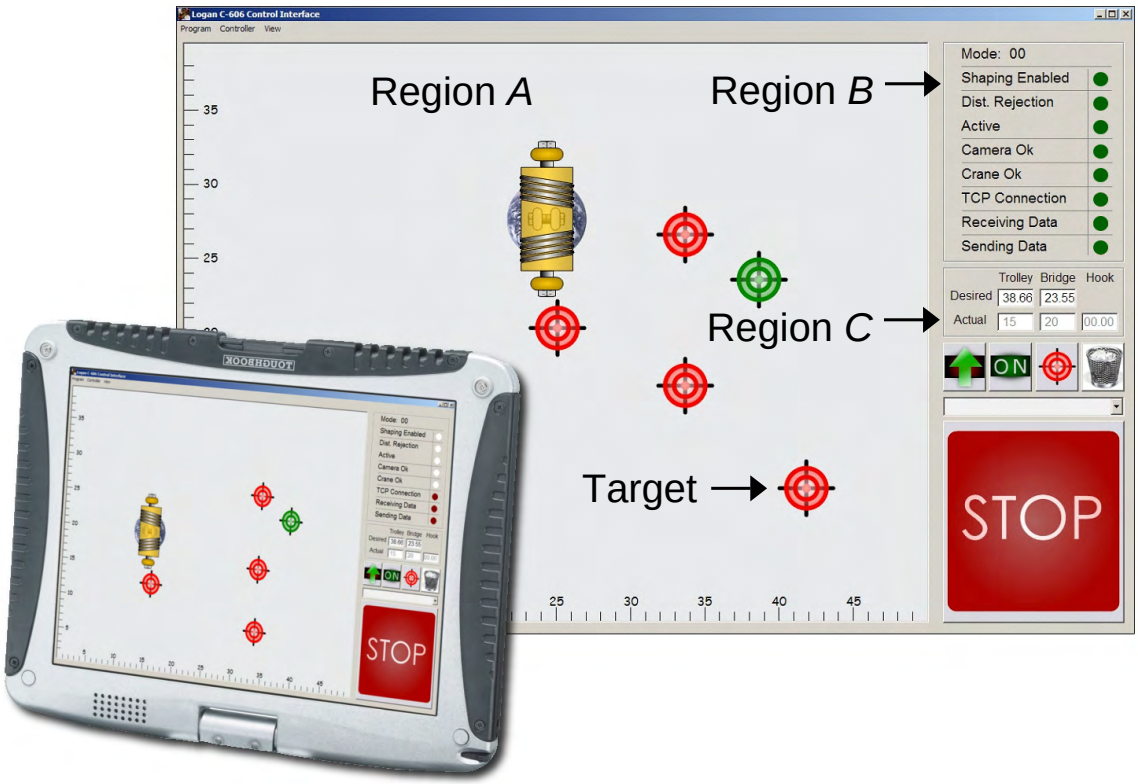


Figure 9.7: Screen Shot of the Logan Crane Visual Interface.

While the visual interface implemented on the Logan crane is very similar to the visual interface described in Chapter 7, one significant distinction exists. The interface of Chapter 7 displays real-time images of a crane and workspace, which are provided by a live video feed. This feature facilitates both local and remote operation because in either scenario, operators are well informed about potential hazards (e.g. obstacles, workers walking through the workspace, etc.). In contrast, the visual interface on the Logan crane merely represents the crane and crane workspace with graphical illustrations, which are updated in real-time in response to crane motion. These illustrations do not provide an operator with the

wealth of visual feedback afforded by a real-time video stream. Consequently, an operator manipulating the Logan crane with the visual interface should be nearby in order to have either a local or mixed perspective.

While the visual interface on the maintenance crane provides less visual feedback than the interface of Chapter 7, this device still provides many other benefits of the originally described visual interface. Commanded motion is still initiated through gesture-based actions on the part of the operator. And, motion trajectories are still generated by a controlling computer. This type of control scenario utilizes intuition-based behavior, which is less complex and requires less cognitive resource use than manual positioning. As a result, operators are more free to direct mental resources toward supervisory tasks, such as monitoring, assessing potential hazards, and payload transport decision making.

9.2.1.2 A Joystick Interface for Simplified Manual Control

Whereas the visual interface facilitates automatic crane motion, the lever and joystick interfaces can be used for manual motion tasks. For some trajectories, the joystick interface permits these types of tasks to be completed more efficiently than the lever interface.

As previously described in Chapter 7, when an operator attempts to command crane motion using a lever interface, he or she mentally accomplishes several steps. First, the desired trajectory of the hook and payload is decoupled into the kinematic components corresponding to the different modes of actuation (i.e. forward-reverse, left-right, up-down). Second, the decoupled trajectories are mentally mapped to the corresponding actuating levers. Finally, the operator attempts to physically depress the correct combination of levers so that the crane traverses along the desired trajectory. For efficient motion, generally two levers must be controlled simultaneously by the operator. Because this is difficult, operators often manipulate the crane along less efficient trajectories that require motion in only one actuation direction at a time.

To manipulate the crane with the joystick, operators point the joystick handle in a desired direction with variable displacement. The direction and displacement of the joystick handle correspond to the direction and velocity of crane travel. This mode of manual control

can facilitate more efficient crane motion than the lever interface. This is because it is easier for an operator using the joystick to simultaneously actuate the crane in both the bridge and trolley directions.

According to the path and interface classification system described in Chapter 8, the *actuation realizability* for the lever interface, for general crane motion, is usually moderate or complex. For the joystick, the actuation realizability classification is reduced to either simple or moderate.

9.2.2 Sensory Information

9.2.2.1 Machine Vision System

Sensory information about hook swing is obtained by using a Siemens 720-series vision system. This camera is a stand-alone image sensor with on-board image acquisition, processing, and communication capabilities. The vision system is mounted on the trolley, near the fulcrum of the hook suspension cables, and oriented to view the hook and surrounding workspace. In this downward-looking configuration, the top of the crane hook is always within the camera field-of-view, as shown in Figure 9.8.

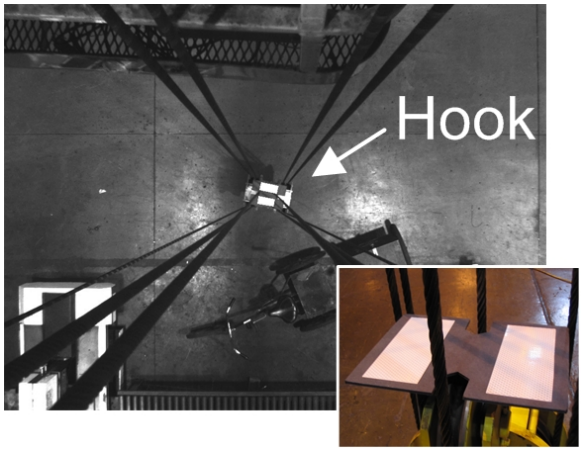


Figure 9.8: Image Captured and Processed by the Machine Vision System (Large Photo). Close-Up Photograph of the Fiducials Mounted on the Hook (Sub Photo).

To facilitate reliable hook tracking, a light-emitting-diode (LED) array was installed. When the vision system acquires an image, the LED array simultaneously pulses (similar to a flash bulb on a camera). Fiducial markers made of retro-reflective material, and mounted

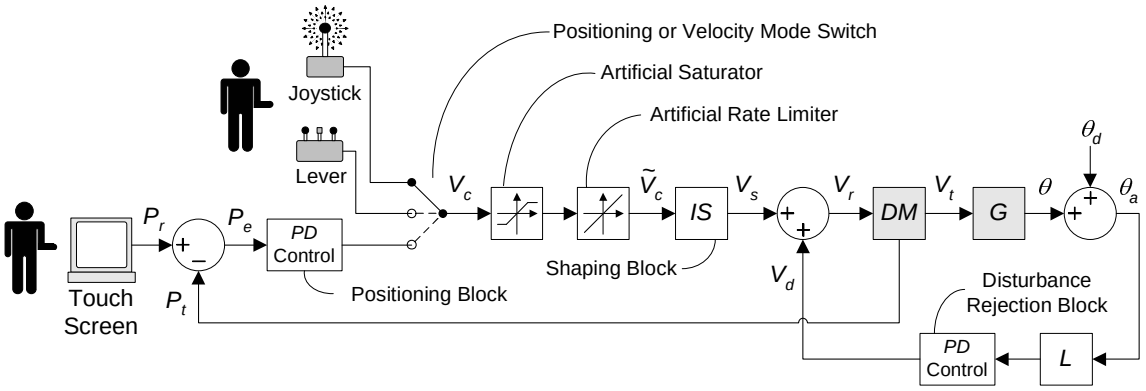


Figure 9.9: Anti-Sway and Positioning Controller.

to the top of the hook, reflect the pulsed light back to the camera lens. By so doing, the fiducial markers are easily discernable from other features in the image. A close-up view of the fiducial markers is shown in the sub photograph of Figure 9.8.

The camera acquires an image every 70 ms. A program within the camera processes the image to obtain coordinate information about the fiducial markers. This information is subsequently communicated to the anti-sway and positioning controller.

9.2.2.2 Laser Range Sensors

Absolute bridge and trolley positions are obtained by using two Banner LT3-series laser range sensors. These position sensors have a range of 50 m and a resolution of approximately 1.2 cm at this distance. Both sensors are mounted on the bridge. One is oriented to detect the position of the trolley along the bridge. The other is oriented to detect the position of the bridge along the stationary runways.

9.2.3 Anti-Sway and Positioning Control

A block diagram of the anti-sway and positioning control is shown in Figure 9.9. This block diagram depicts a control architecture wherein the original crane system is integrated into a two-loop feedback structure. The control generates reference velocity commands that, when issued to the nonlinear drive and motors plant, DM, achieve three desirable results in the payload: 1) precise positioning capability, 2) motion-induced oscillation suppression, and 3) disturbance rejection.

In the subsections that follow, brief descriptions of the control elements that enable these result to be achieved are provided. The reader interested in the development of this control architecture and the results of rigorous stability analyses should refer to [111, 108, 98].

9.2.3.1 Motion-Induced Oscillation Suppression

Motion-induced oscillation suppression is achieved by incorporating a ZV input-shaping filter into the signal path of the maintenance crane. In Figure 9.9, the input shaper is represented by the block labeled IS. This shaper is designed to negate the oscillatory dynamics of the closed-loop transfer function formed by the disturbance rejection feedback loop. The input shaper accepts commanded velocity signals, $\widetilde{V}_c$, which originate from one of three elements. If the crane is being manually manipulated, then signals from the joystick or lever interface are issued to the input shaper. If the crane is being automatically positioned through the use of the touchscreen, then signals from a positioning control block are issued to the shaper. The shaped commands, V_s, are then used as reference commands for the disturbance rejection loop.

While input shaping works effectively for linear systems, Chapter 3 described how the saturation and rate limiting behaviors exhibited by the drives and motors can significantly reduce the capacity of the shaped commands to eliminate oscillation. Therefore, two strategies presented in Chapter 4 for mitigating these detrimental effects were incorporated into the control system. An artificial saturation element, and an artificial rate limiting element were inserted into the signal path such that these elements filter reference commands before they are modified by the input shaper. The saturation and rate limiting parameters of the artificial elements were selected using the fixed saturation and fixed rate limiting techniques of Section 4.1.1. By so doing, the shaped commands generated by the input shaper are not corrupted by the saturation and rate limit elements contained in the drives and motors.

It should be noted that the saturation and rate limiting mitigation techniques of Chapter 4 were presented in the context of mitigating a single nonlinearity. Here, however, both the fixed rate limit and fixed saturation constraints are implemented in a series architecture. It may be easily shown that combining the constraints in this manner enforces both the rate

limiting and saturation mitigation conditions of Section 4.1.1. Therefore, the detrimental effects of the inherent saturator and rate limiter within the drives and motors are negated. The effectiveness of this mitigation technique at enabling the input shaper to suppress oscillation is demonstrated in the following section, which describes the positioning portion of the control architecture.

9.2.3.2 Precise Payload Positioning

Consider the transfer function in (9.2) that relates the cable angle of the crane to the velocity of the trolley. A state-space representation of the plant follows directly from the coefficients of the transfer function:

$$\dot{\mathbf{q}} = A\mathbf{q} + \mathbf{b}v_t(t), \qquad \theta(t) = \mathbf{cq} \tag{9.4}$$

where,

$$A = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \qquad \mathbf{c} = \begin{bmatrix} 0 & \frac{-\omega_n^2}{g} \end{bmatrix} \tag{9.5}$$

and $\mathbf{q} = [q_1\ q_2]^T$ is the state vector.

Given the relationship between q_2 and θ established in 9.4, one may recognize that the state q_2 is equal to $-\theta L$, the relative horizontal displacement between the trolley and the payload.

Computing the eigenvalues of A yields:

$$\lambda = -\zeta\omega_n \pm \omega_d j \tag{9.6}$$

Because the real part of the eigenvalues of A are negative, the state q_2 is asymptotically stable in the sense of Lyapunov [12]. Therefore, q_2 always approaches zero. By this formal treatment of the state equations, an obvious fact is emphasized: the payload always come to rest directly beneath the trolley. Therefore, precise positioning of the trolley is equivalent to precise positioning of the payload. The architecture of the anti-sway and positioning controller exploits this system behavior. Precise payload positioning is accomplished by using position information from the laser range sensors to control the position of the trolley.

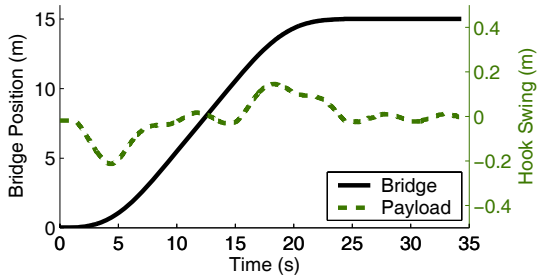
P_r, P_t, and P_e are the reference trolley position, actual trolley position, and positioning error, respectively.

In response to the positioning error, a proportional-derivative (PD) control block generates a velocity signal that attempts to drive the crane toward the desired location. If this signal were issued *directly* to the drives and motors, then the objective of trolley positioning would be accomplished, but, noticeable hook swing would be exhibited. However, because the input shaper, in conjunction with the artificial saturator and rate limiter, filter the commanded velocity signal, the dual objective of driving the crane toward a desired position, while also preventing motion-induced oscillation is achieved.

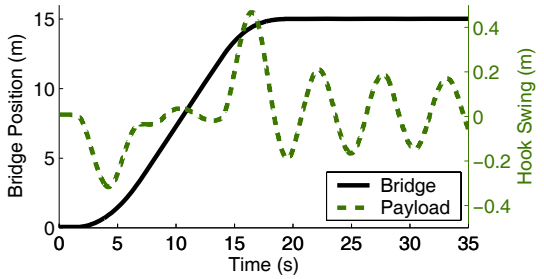
To demonstrate the effectiveness of filtering the commanded velocity signal with an input shaper, artificial saturator, and artificial rate limiter, consider Figure 9.10(a). This figure depicts the response of the bridge (solid line) and payload (dotted line) to a position command of 15 m. The left vertical axis of this figure represents the displacement of the bridge throughout the positioning move. The right vertical axis represents the horizontal displacement of the hook throughout the positioning move. For this trial, the PD gains of the disturbance rejection control block were set equal to zero. By so doing, the oscillation suppression exhibited by the system could be entirely attributed to the input shaper and mitigation techniques. The proportional and derivative gains of the positioning control block were set equal to 0.3 and 0.25, respectively. The input shaper was tuned to cancel the oscillatory poles of G.

For comparison purposes, Figure 9.10(b) shows the crane response from an identical trial, except the artificial saturator and rate limiter were removed from the control architecture. The detrimental effects of actuator nonlinearities are exhibited by the oscillatory response of the payload.

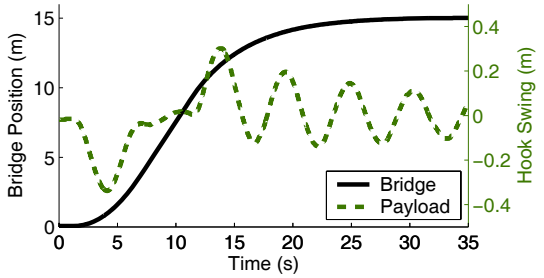
Figure 9.10(c) shows the results when the input shaper, as well as both of the mitigation blocks were removed. This scenario reduces the system to a conventional PD feedback controller. In the system, there is no mechanism for eliminating the oscillation of the payload. This is evident in the oscillatory response of the payload.



(a) Positioning Control with Input Shaping, and Mitigation.



(b) Positioning Control wit Input Shaping, but without Mitigation.



(c) Positioning Using Conventional PD-Control.

Figure 9.10: Positioning and Oscillation Suppression Capability of Different Controllers.

The experimental results of Figure 9.10 demonstrate that input shaping, used in conjunction with mitigation, is effective for motion-induced oscillation suppression, even on systems with saturation and rate limiting nonlinearities.

The use of input shapers within feedback loops is a non-conventional control architecture called *closed-loop signal shaping* (CLSS). A moderate body of literature exists that addresses the stability of these types of systems [125, 37, 112, 30], and their beneficial properties [32]. For CMS-enabled positioning control, stable PD control gains that achieved desirable system performance were obtained by using a root locus approximation technique presented in [30]. This root locus technique neglects the nonlinear elements within the control structure in order to facilitate a simplified analysis. Consequently, this analysis does not guarantee the stability of the nonlinear system. It does, however, provide insight into the dynamics of the CLSS system, and reveals the anticipated behavior when the system operates within its linear region. The PD control gains obtained from the root locus approximation technique were rigorously verified on a nonlinear model of the system by using numerical methods presented in [108].

9.2.3.3 Disturbance Rejection

While the primary source of cable sway is commanded motion, secondary sources of oscillation, such as external disturbances or unmodeled dynamics, also contribute to undesired hook swing. These types of disturbances may be modeled as inducing a disruptive angle, θ_d, that is summed with the undisturbed angle, θ, to produce the actual cable angle, θ_a.

Disturbance rejection is accomplished by making use of the machine vision system to provide sensory feedback of the actual cable angle, θ_a. For small angles, the horizontal displacement of the hook from the vertical at-rest position can be reasonably estimated by $L\theta_a$. This quantity is utilized in the disturbance rejection PD block, which generates corrective velocity commands to damp out the disruptive oscillations. These signals are added to the shaped velocity signals, V_s, obtained from the input shaper. The combined signal is then issued to the system drives.

Although the drives and motors contain nonlinear elements, these components do operate within their linear regions frequently. Therefore, during these periods of time, superposition holds. Thus, because the signal V_s continually drives the crane toward a desired set point, and the signal V_d attempts to damp out oscillation, the CMS control can achieve the dual objectives of positioning and disturbance rejection. Furthermore, because V_s is an input-shaped command, motion-induced hook swing will be reduced. In this way, the CMS control eliminates motion-induced oscillation, rejects disturbances, and enables precise positioning of the payload.

9.2.4 Beneficial Attributes of Combining Input Shaping with Feedback Control

The control architecture presented in the previous subsection allocates the task of motion-induced oscillation suppression to the input shaping filter. Disturbance-induced oscillation suppression is achieved through feedback control. While feedback *alone* could suppress both motion and disturbance-induced payload swing, beneficial system behavior is exhibited when these tasks are separated [57, 39].

Input shaping reduces motion-induced oscillation in an *anticipatory* manner, as opposed to the *reactive* manner of feedback control. Oscillation suppression is accomplished with the

reference signal that anticipates the oscillation before it occurs, rather than with a *correcting* signal that attempts to restore deviations back to a reference signal. In the context of crane control, this means that motion-induced oscillation can usually be suppressed within one-half period of oscillation when using input shaping. To achieve similar performance with feedback control, aggressive gains must be used, which results in higher actuator effort, and greater command distortion than when using input shaping.

Another beneficial consequence of allocating motion-induced oscillation suppression to an input shaping filter is related to multi-mode cable sway. For some payload and rigging configurations, cranes can exhibit "double-pendulum" oscillatory dynamics [85]. In such cases, aggressive disturbance rejection feedback gains will cause the nonlinear drives and motors to exhibit limit cycling. By utilizing input shaping, motion-induced oscillation suppression of the multi-mode system can be quickly suppressed without aggressive control gains.

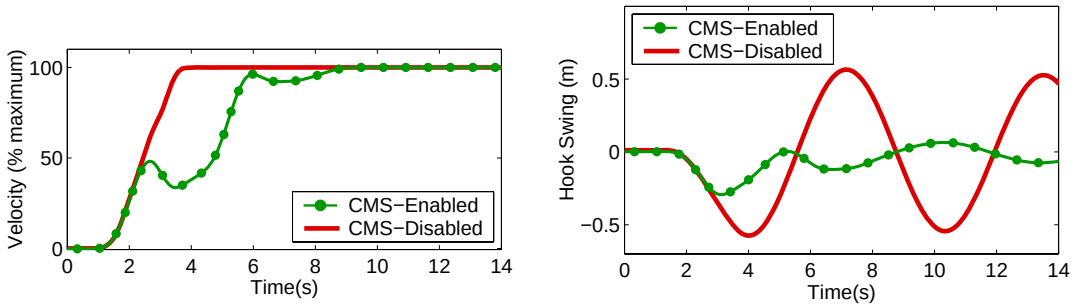
9.3 Performance Evaluation

The anti-sway and positioning capabilities of the CMS-equipped maintenance crane were rigorously tested. Section 9.3.1 discusses motion-induced oscillation suppression. Section 9.3.2 discusses disturbance rejection. Finally, Section 9.3.3 discusses positioning.

9.3.1 Motion-Induced Oscillation Suppression

To evaluate the capabilities of the CMS to suppress motion-induced oscillation, the crane was driven both with and without the CMS enabled. In the first set of such tests, the lever interface was used to issue a step command in velocity to the bridge. The velocity response of the bridge to this command is shown with the solid line in Figure 9.11(a). The oscillatory response of the hook to the bridge motion is shown with the solid line of Figure 9.11(b). The same test was conducted when the CMS was enabled. These results are shown with the dotted lines of Figure 9.11. The bridge exhibits a noticeably different motion profile that causes much less hook swing.

Similar results were obtained when these tests were repeated in the trolley direction, and simultaneously in the trolley and bridge directions. The bar graph of Figure 9.12



(a) Velocity Commands.

(b) Hook Swing.

Figure 9.11: Motion-Induced Oscillation Suppression with and without the CMS.

summarizes the swing amplitude results from each of these tests.

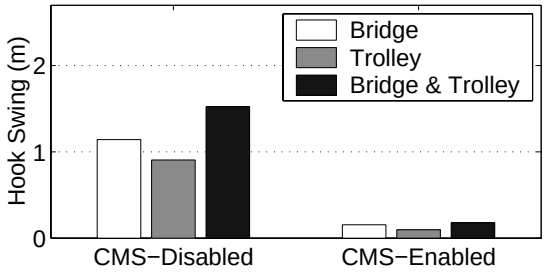


Figure 9.12: Residual Hook Swing Amplitude in Response to a Velocity Step Command.

Another set of tests were conducted where a pulse in velocity (instead of a step) was issued to the crane. For each of these tests, the lever interface was used to quickly accelerate the crane to maximum velocity. After the crane had been traveling at maximum velocity for approximately two seconds, the crane was quickly decelerated to a full stop. The swing amplitude results of these tests are summarized in the bar graph of Figure 9.13.

The results summarized in Figures 9.12 and 9.13 demonstrate that the CMS-equipped crane can reduce motion-induced oscillation by roughly 90%.

9.3.2 Disturbance Rejection

The disturbance rejection capabilities of the CMS were evaluated by imparting an initial hook swing to the system while the CMS was deactivated. Then, the CMS was activated while crane position and hook position data were recorded. This experiment was repeated at several different suspension cable lengths ranging from 10 m to 3 m.

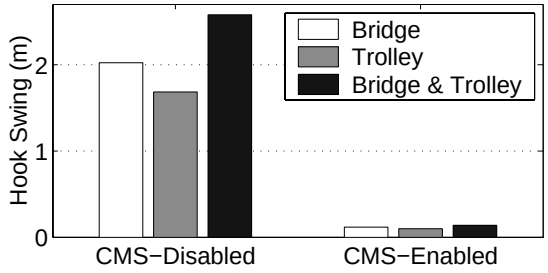


Figure 9.13: Residual Hook Swing Amplitude in Response to a Velocity Pulse Command.

A typical system response to this experiment is shown in Figure 9.14. The trolley position is shown with the solid line and the payload position is shown with the dotted line. The CMS was activated at approximately time $t = 6$ s. The hook oscillations were mostly damped out by time $t = 17$ s, a time duration of approximately two oscillation periods.

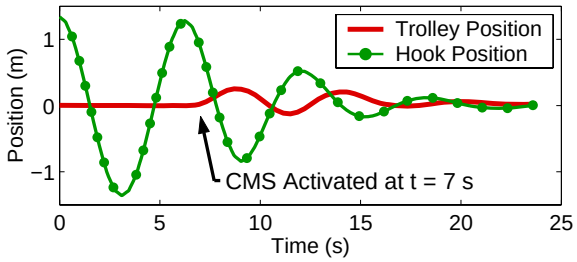
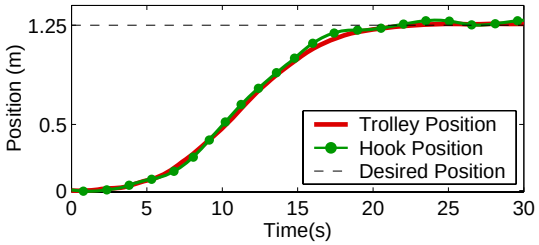


Figure 9.14: Cancelation of Disturbance-Induced Oscillations.

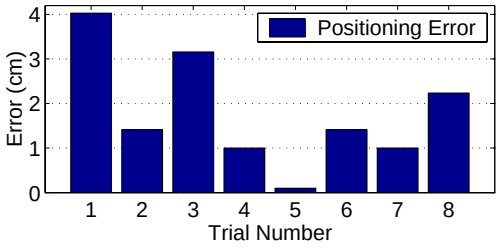
9.3.3 Positioning Capabilities

The positioning capabilities of the CMS were evaluated by issuing several reference positions to the crane. The automatic control attempted to drive the crane to the desired positions while limiting motion-induced and disturbance-induced oscillations. The response of the crane to a typical position command is shown in Figure 9.15(a). This figure depicts motion in the trolley direction only, but each test required the crane to be positioned simultaneously in both the trolley and bridge directions. For the trial depicted in Figure 9.15(a), the trolley started at the 0 m location and was then commanded to move the the 1.25 m location. The solid curve represents the position of the trolley throughout the experiment; the dotted curve represents the position of the hook.

The positioning error results for eight trials are shown in Figure 9.15(b). The vertical



(a) CMS-Enabled Trolley Response to a Position Command of 1.25 m.



(b) Final Positioning Error Measured Radially from the Desired Position to the Actual Crane Position.

Figure 9.15: Positioning Performance of the CMS-Enabled Maintenance Crane.

axis of this figure represents the final radial positioning error between the desired crane position and the actual crane position. Based upon these results, the average radial positioning error is approximately 1.8 cm with a standard deviation around the average equal to approximately 1.3 cm.

9.4 Conclusion

By using the technological developments presented earlier in this thesis, a crane manipulation system (CMS) was developed and implemented on a 30-ton bridge crane. The system is comprised of a machine vision system for sensing hook position, laser range sensors for obtaining bridge and trolley positions, a joystick interface for simplifying gross motion tasks, a visual touchscreen interface for simplifying positioning tasks, and an anti-sway/positioning control law. The CMS enables operators to manipulate the crane in a nearly swing-free manner. Motion-induced oscillations of the hook were reduced by roughly 90%. The control system can also reject externally-induced hook swing. Precision positioning of the hook was demonstrated with positioning capability on the order of a few centimeters. The advantageous functionality of the CMS can be largely attributed to advancements made in the areas of nonlinear input-shaping theory and human/crane interaction.

CHAPTER X

CASE STUDY: ANTI-SWAY AND POSITIONING CONTROL OF A 35-TON CRANE WITH MULTI-MODE DYNAMICS[1]

CHAPTER SUMMARY: *The detrimental effects of payload swing on operational efficiency are compounded when a crane exhibits multiple modes of oscillation. A case study is presented in which a 35-ton crane that exhibits multi-mode oscillation is equipped with the previously described crane manipulation system (CMS). A thorough analysis of the crane's oscillatory behavior is presented and used to design the input shaping component of the CMS.*

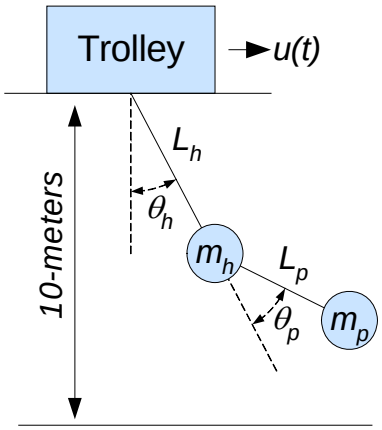
Figure 10.1(a) shows a photograph of a 35-ton crane located at Logan Aluminum, a manufacturer of aluminum sheet products. This crane is used to transport cylindrical-shaped "rolls" to different machining and storage locations. The precision with which operators are required to position the rolls is generally on the order of ± 5 *cm*.

The payload and rigging configuration used to transport the rolls cause the system to exhibit noticeable "double-pendulum" behavior. For example, a typical payload response to crane motion is depicted in Figure 10.2. This plot shows the response of the hook when a human operator manipulated the crane from one position to another. Two oscillatory modes are clearly discernable. This type of payload swing is much more difficult for operators to suppress than single-mode, pendulum-like motion [85]. Consequently, multi-mode payload swing also compounds the difficulty and safety of positioning tasks.

In order to suppress the multiple vibration modes, and to simplify positioning tasks, the crane manipulation system (CMS) previously described in Chapter 9 has been implemented on the roll-shop crane. In Chapter 9, the CMS was described in the context of its implementation on a 30-ton maintenance crane, which exhibited predominantly single-mode payload

[1]Portions of this chapter are published in [105, 104].

(a) 35-Ton Industrial Crane Shown with Rigging and Payload.



(b) Lumped-Mass Model of the Crane.

Figure 10.1: Industrial Crane and Crane Model.

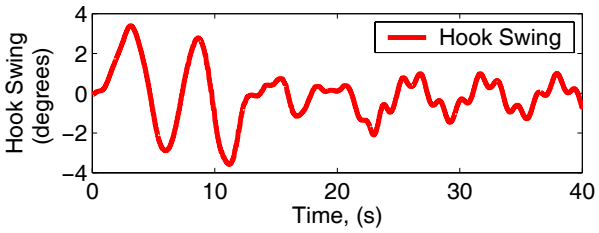


Figure 10.2: Multi-Mode Behavior of Hook in Response to Bridge Motion.

dynamics. The difference between the oscillatory behaviors of the maintenance and roll-shop cranes constitute an important consideration in the design of the CMS. Consequently, the objective of this chapter is to expand the application space of the CMS by describing how this system can be utilized on cranes with multiple modes of payload swing. More specifically, this chapter seeks to:

1. *Present a thorough dynamic analysis of the roll-shop crane in order to obtain ranges of anticipated oscillation frequencies.*

2. *Incorporate the multi-mode dynamic properties of the roll-shop crane into the design of the CMS.*

Section 10.1 presents a dynamic analysis of the roll-shop crane in order to determine the ranges of the anticipated payload-swing frequencies. Section 10.2 discusses how information

from the frequency analysis is used to design critical aspects of the CMS. Experimental results from the crane are presented in Section 10.3.

10.1 Frequency Analysis

To implement the desired control, estimates of the oscillation frequencies are necessary. This information can be determined from a dynamic analysis of the system model previously illustrated in Figure 10.1(b). This models captures the "double-pendulum" behavior of the rigging configuration used with the roll-shop crane.

The symbol, $u(t)$, is the acceleration of the trolley. The parameters m_h, L_h, and θ_h, are the mass, suspension length, and angular displacement associated with the hook. The remaining parameters are associated with the payload. During operation of the crane, these parameters change between their nominal maximum and minimum values, according to Table 10.1.

Table 10.1: Nominal Operating Range for Model Parameters.

	Minimum	**Maximum**
L_h	3 m	9 m
L_p	0.5 m	4 m
$M = m_p/m_h$	0	100

If it is assumed that the cable and rigging length are static or change slowly, then the linearized equations of motion can be expressed as [85]:

$$\ddot{\theta}_h(t) + \left(\frac{g}{L_h}\right)\theta_h - \left(\frac{gM}{L_h}\right)\theta_p = -\frac{u(t)}{L_h} \tag{10.1}$$

$$\ddot{\theta}_p(t) - \left(\frac{g}{L_h}\right)\theta_h + \left(\frac{g}{L_p} + \frac{gM}{L_p} + \frac{gM}{L_h}\right)\theta_p = \frac{u(t)}{L_h} \tag{10.2}$$

Closed-form expressions for the oscillatory frequencies can be obtained from (10.1) and (10.2) as:

$$\omega_{1,\,2} = \sqrt{\frac{g}{2}}\sqrt{(1+M)\left(\frac{1}{L_h} + \frac{1}{L_p}\right) \mp \beta} \tag{10.3}$$

where g is the acceleration due to gravity, and

$$\beta = \sqrt{(1+M)^2\left(\frac{1}{L_h} + \frac{1}{L_p}\right)^2 - 4\left(\frac{1+M}{L_p L_h}\right)} \tag{10.4}$$

10.1.1 Low-Mode Frequency Range

The range over which the low-mode varies can be calculated by assuming that the combined mass of the payload plus the hook is concentrated at either the hook or the payload. In either case, the oscillatory behavior of the system mimics that of a single pendulum. In this situation, the frequency is given by:

$$\omega_1 = \sqrt{g/L_{total}} \tag{10.5}$$

The symbol, L_{total}, is the distance from the fulcrum of the suspension cable to the center of mass. If the mass is concentrated at the hook, then L_{total} is equal to L_h. If the mass is concentrated at the payload, then L_{total} is equal to $L_h + L_p$. Using (10.5), and given the parameter variation range specified in Table 10.1, the anticipated frequency range for the low mode is approximately 1 rad/s to 1.8 rad/s.

10.1.2 High-Mode Frequency Range

The frequency of high-mode excitation depends on the two suspension lengths, L_p and L_h, and the mass ratio, M. Given the anticipated variance of these parameters from Table 10.1, a range of anticipated high-mode frequencies can be obtained directly from (10.3). To accomplish this, (10.3) was used to generate the two surfaces plotted in Figure 10.3. These surfaces represent the value of the high-mode frequency as the mass ratio, M, and rigging length, L_p, are varied throughout their expected ranges. The upper surface corresponds to a payload/rigging configuration when L_h is 9 m. The lower surface corresponds to a payload/rigging configuration when L_h is 3 m.

Figure 10.3 indicates that the high-mode frequency can vary over a much larger range than the low-mode frequency. The minimum frequency value indicated on this figure is near 1 rad/s, while the highest frequency value is approximately 50 rad/s. However, the control only needs to suppress those frequencies that noticeably contribute to the system response.

To determine the contribution of the second mode to hook displacement, the response of the hook is expressed terms of components arising from ω_1 and ω_2 [85]. The angular

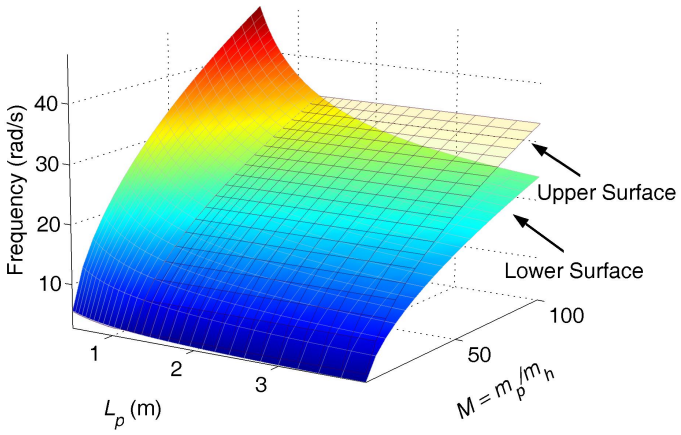


Figure 10.3: High-Mode Frequency Range for Extrema Payload/Rigging Configuration. Upper Plane ($L_h = 9\ m$). Lower Plane ($L_h = 3\ m$).

response of the hook to an impulse, A, is:

$$\theta_h(t) = C_{low}\sin(\omega_1) + C_{high}\sin(\omega_2) \tag{10.6}$$

where,

$$C_{low} = \frac{A\omega_1(1+\omega_2^2 L_h\alpha)}{\beta L_h g} \tag{10.7}$$

$$C_{high} = -\frac{A\omega_2(1+\omega_1^2 L_h\alpha)}{\beta L_h g} \tag{10.8}$$

and,

$$\alpha = \frac{-g(1+M)}{\omega_1^2\omega_2^2 L_h L_p} \tag{10.9}$$

The coefficients, C_{low} and C_{high}, directly influence the displacement contributions from the low and high modes. Therefore, one may determine the relative contribution of the high mode to the overall response by examining the ratio, C_{high}/C_{low}. The range of frequencies over which this this ratio is less than an acceptable threshold can be neglected.

In Figure 10.4, the modal contribution ratio, C_{high}/C_{low}, is plotted when L_h is set equal to 3 m (the surface generated when L_h equals 9 m is negligibly small). Additionally, a contour line has been added to this figure to indicate the parameter ranges where the high mode contributes more than 15% to the total hook displacement. The contour line partitions the parameter space into two regions. In one region, the corresponding high-mode

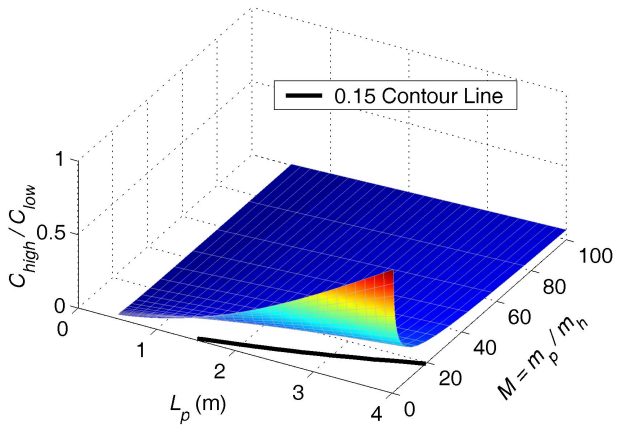


Figure 10.4: Modal Contribution to Hook Response.

frequencies can be neglected; in the other region, the corresponding high-mode frequencies should be considered in the design of the control.

In light of Figure 10.4, the large range of high-mode frequencies previously identified in Figure 10.3 can be significantly reduced. Considering Figure 10.4 together with Figure 10.3 indicates that the non-negligible frequencies occur when the mass ratio, M, is less than 20, and when the rigging length, L_p, is greater than 2 m. Therefore, based on the overall shape of the surfaces in Figure 10.3, a good estimate of the non-negligible high-mode frequency range can be obtained by looking at a cross section of Figure 10.3 when L_p is equal to 4 m, and the mass ratio varies between 0 and 20. This cross section is shown in Figure 10.5. From this figure, one may conclude that the non-negligible high-mode frequency ranges from approximately 2 rad/s to 15 rad/s.

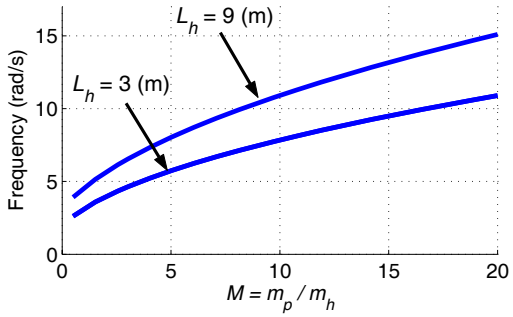


Figure 10.5: Non-Negligible High-Mode Frequency Range.

The estimated frequency ranges for both the high and low modes of oscillation are

summarized in Table 10.2.

Table 10.2: Anticipated Frequency Ranges.

	Minimum	**Maximum**
ω_1	1 rad/s	1.8 rad/s
ω_2	2 rad/s	15 rad/s

10.2 CMS Input Shaper Design

The primary attribute of the CMS is that it suppresses hook swing during manual and automatic positioning operations. This is accomplished with a correctly designed input-shaping filter. To suppress hook swing over the range of anticipated frequencies determined in Section 10.1, the amplitudes and time locations of the impulses comprising the input shaper were determined by solving the constraint equations of Section 1.2.1. Recall that these constraints limit the payload oscillation to be less than a tolerable vibration threshold, V_{tol}.

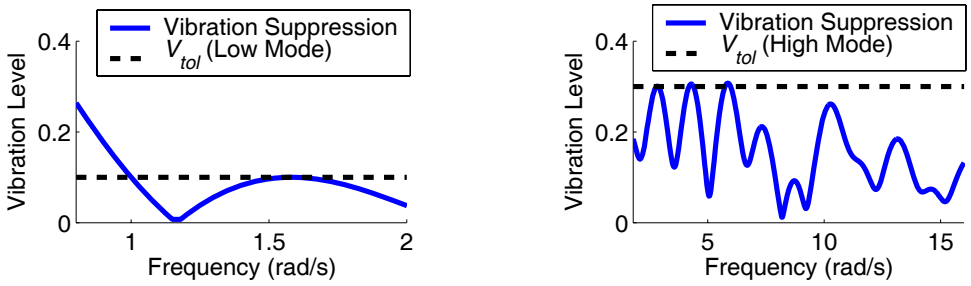
Generally speaking, for most of the payload/rigging configurations utilized with the roll-shop crane, the low-frequency oscillations dominate the hook response. Accordingly, V_{tol} for the low frequency range was set equal to 0.1. This implies that with input shaping, the low frequency response of the hook will be less than 10% of the response without input shaping. This reduced level of vibration is sufficient for significantly improving operational performance. From Figure 10.4, it can been seen that the relative contribution of the high frequency mode to the overall response is a fraction of the contribution from the low frequency mode. Therefore, V_{tol} for the high frequency range was set higher than V_{tol} for the low mode, at a value of 0.3. By increasing the value of V_{tol} for the second mode, the duration of the input shaper can be reduced [55]. This is desirable because the system can respond more rapidly to issued commands when the duration of the input shaper is short.

The values of the impulse amplitudes, A_i, and times, t_i, that satisfy the constraint equation throughout the anticipated frequency ranges, and at the specified levels of V_{tol}

were calculated as[2]:

$$\begin{bmatrix} A_i \\ t_i(s) \end{bmatrix} = \begin{bmatrix} 0.42 & 0.15 & 0.10 & 0.12 & 0.10 & 0.11 \\ 0 & 1.35 & 1.94 & 2.46 & 3.22 & 4.30 \end{bmatrix} \tag{10.10}$$

The frequency vibration suppression properties of the designed input shaper are graphically represented in Figures 10.6(a) and 10.6(b). The solid lines indicate the amplitude of vibration that would be induced into a system by a command shaped with the designed input shaper. This quantity is represented in the figure as a fraction of the oscillation that would be induced by the unshaped counterpart. The tolerable levels of vibration, V_{tol}, for each frequency range are shown with the dashed lines. Note that the vibration levels remain at or below these values throughout the desired ranges.



(a) Low-Mode Frequency Range.

(b) High-Mode Frequency Range.

Figure 10.6: Vibration Suppression Properties of the Input Shaper Over Low and High-Mode Frequency Ranges.

10.3 Performance Evaluation

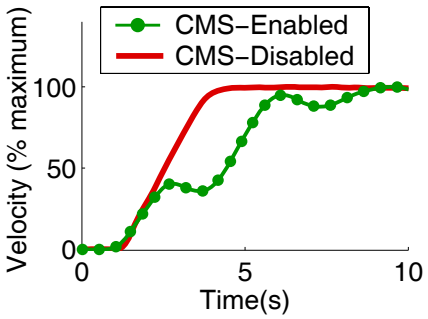
The anti-sway and positioning capabilities of the roll-shop crane were tested in a similar manner as the maintenance crane of Chapter 9. The following sections present the results from these tests.

[2]The complete parameter set used to calculate the derived shaper include both the low and high frequency range: 1 - 1.8 rad/sec and 2 - 15 rad/sec, respectively; the tolerable levels of vibration for the low and high modes: 0.1 and 0.3 respectively; and the damping ratio for the low and high modes: 0.2 and 0.05, respectively. The damping ratio for the low mode is based on the level of damping added by the disturbance rejection component of the CMS.

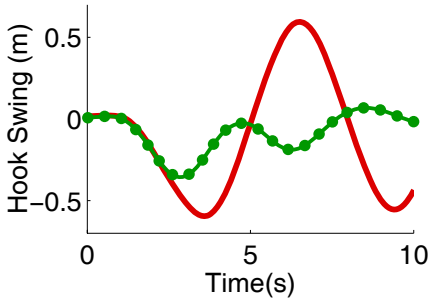
10.3.1 Motion-Induced Oscillation Suppression

To evaluate the capabilities of the CMS to suppress motion-induced oscillation, the crane was driven both with and without the CMS enabled. In the first set of such tests, the CMS was disabled. Then, a step command in velocity was issued to the system while the hook swing in the bridge direction was measured. The same test was then repeated with the CMS enabled.

The response of the crane to this test is shown in Figure 10.7. The velocity response of the bridge is shown in Figure 10.7(a). The motion of the hook is shown in Figure 10.7(b). The solid lines represent the bridge and hook motion when the CMS was disabled, the dotted lines represent the bridge and hook motion when the CMS was enabled. Note that when the CMS was enabled, the bridge exhibited a noticeably different motion profile that caused much less hook swing.



(a) Velocity Response of the Crane.



(b) Hook Swing Response of the Crane.

Figure 10.7: Motion-Induced Oscillation Suppression with and without the CMS.

Similar results were obtained when these tests were repeated in the trolley direction, and simultaneously in the trolley and bridge directions. Figure 10.8 summarize the swing amplitude results from each of these tests. The vertical axis of this graph represents the amplitude of residual hook swing.

Another set of tests were conducted where a pulse in velocity (instead of a step) was issued to the crane. For each of these tests, the crane was quickly accelerated to maximum velocity. After the crane had been traveling at maximum velocity for approximately four seconds, the crane was quickly decelerated to a full stop. The swing amplitude results of

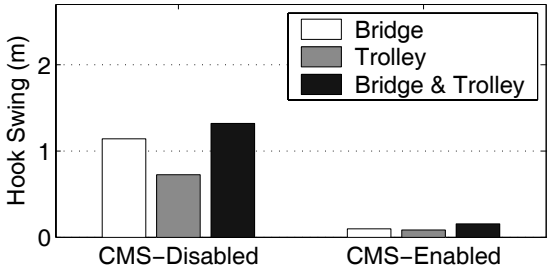


Figure 10.8: Residual Hook Swing Amplitude in Response to a Velocity Step Command.

these tests are summarized in the bar graphs of Figure 10.9.

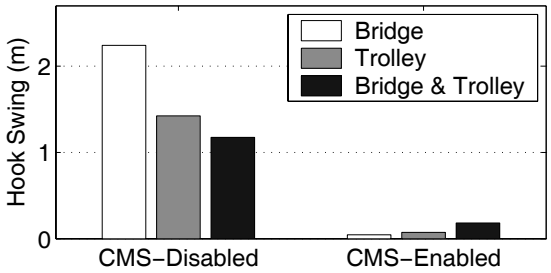


Figure 10.9: Residual Hook Swing Amplitude in Response to a Velocity Pulse Command.

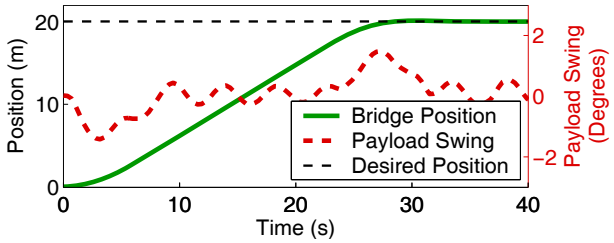
The results summarized in Figures 10.8 and 10.9 demonstrate that the CMS-equipped, multi-mode crane can reduce motion-induced oscillation by roughly 90%.

10.3.2 Positioning Capabilities

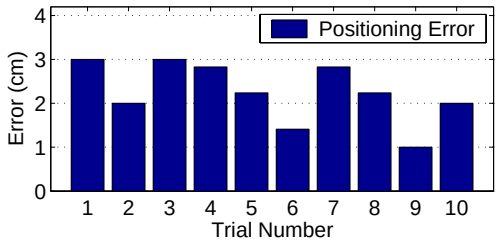
The positioning capabilities of the CMS were evaluated by issuing several reference positions to the roll-shop crane. The automatic control drove the crane to the desired positions, while limiting payload oscillation. The final positioning error was recorded for each trial.

The response of the crane to a typical position command is shown in Figure 10.10(a). This figure depicts motion in the bridge direction only, but each test required the crane to be positioned simultaneously in both the trolley and bridge directions. For the trial depicted in Figure 10.10(a), the bridge started at the 0 m location and was then commanded to move to the 20 m location. The solid curve represents the position of the bridge throughout the experiment; the dashed curve represents the oscillation of the hook. The peak-to-peak oscillation exhibited during this motion represents less than 20 cm of horizontal payload displacement. Without the CMS control, typical horizontal displacement is on the order of

100 cm to 200 cm.



(a) CMS-Enabled Bridge Response to a Position Command of 20 m.



(b) Final Positioning Error Measured Radially from the Desired Position to the Actual Crane Position.

Figure 10.10: Positioning Performance of the CMS-Enabled Roll-Shop Crane.

The positioning capabilities of the CMS-enabled crane is summarized in Figure 10.10(b). The vertical axis of this figure represents the final radial positioning error between the desired crane position and the actual crane position. Based upon these results, the average radial positioning error for the crane is approximately 2 cm with a standard deviation around the average of approximately 1 cm.

10.4 Conclusion

A crane manipulation system (CMS) was implemented on an industrial bridge crane that exhibits significant multi-mode dynamics. A dynamic analysis identified the frequency ranges that noticeably contribute to the oscillatory behavior of the system. An input-shaping filter was designed and used with the CMS to suppress oscillation over the anticipated range of frequencies. With the designed input shaper, the CMS enabled operators to manipulate the crane in a nearly swing-free manner. Motion-induced oscillations of the hook were reduced by roughly 90%. Precision positioning of the hook was demonstrated with a positioning capability of a few centimeters.

CHAPTER XI

RESEARCH SYNOPSIS AND EXTENSIONS

Industrial productivity is inexorably linked with effective control of human operated machines. Two diverse, yet substantially relevant aspects of human-assisted control are related to how operators issue commands to machines, and how those machines respond to issued commands. Effective management of these two areas can significantly influence safety, efficiency, and ease-of-use. Consequently, the research thrusts of this thesis have been directed toward the fields of control theory and human/machine interaction. These investigations were conducted in the context of industrial crane control.

11.1 Nonlinear Input Shaping

Part one of this thesis investigated input-shaping control of flexible machines subject to actuator nonlinearities. When machines behave in a linear or nearly linear manner, input shaping virtually eliminates motion-induced oscillation. Unfortunately, nonlinear behavior, such as saturation and dead-zone, are ubiquitous within every facet of industry. In the past, when input shaping has been implemented on machines with significant actuator nonlinearities, unanticipated and undesirable system behavior has resulted. This is because the procedure for formulating input-shaping controllers generally oversimplifies the actuation process.

In light of this deficiency, efforts were directed toward the goal of understanding how nonlinear actuation affects input shaping. This was accomplished by developing the deconvolution analysis technique, which provides quantitative and qualitative descriptions of command-induced oscillation. The analysis tool was subsequently used to predict the effects of saturation, rate limiting, dead-zone, and backlash on input-shaped systems. This endeavor produced $\boldsymbol{R}$-value plots - graphical constructs that concisely describe the influence of nonlinear elements on given input shapers. The detrimental effects of nonlinear elements

were quantified with the $\boldsymbol{R}$-value plots in terms of the oscillation produced by the nonlinearity in the controlled plant. It was demonstrated that although saturation, rate limiting, dead-zone, and backlash can significantly inhibit input shapers from suppressing vibration, the levels of exhibited vibration are generally less than those obtained when input shaping is not used.

Because actuator nonlinearities inhibit vibration suppression, methods for mitigating the oscillatory effects of saturation, rate limiting, dead-zone, and backlash were developed. Mitigating oscillation caused by saturation or rate limiting involves conditioning input commands with artificial saturation or rate limiting elements. Two methods for selecting the parameters associated with the artificial elements were presented. Variable parameter mitigation permits rapid motion of the controlled system; fixed parameter mitigation is simple to implement. Mitigating oscillation caused by dead-zone or backlash involves using inverted models of these elements. Because inversion methods tend to be very sensitive to modeling errors, the robustness of the approach to parameter uncertainty was analyzed. It was demonstrated that the inversion approach is effective, even when significant parameter uncertainty exists.

Finite state input shaping is concerned with the implementation of input shaping on system that can only be propelled at discrete levels of actuator effort. Such systems can produce oscillation in response to shaped commands. The cause of this undesirable behavior is that input shapers can generate non-admissible states, which are unrealizable by the discrete actuators. To better understand how input shaping can be implemented on these types of systems, an analytical approach for identifying non-admissible states was presented. The non-admissible state distribution was used to predict how finite-state input shaping affects operational performance.

The phenomenon of overcurrenting was investigated in the context of the command space. The command space provides insight into how overcurrenting occurs, how it can be mitigated, and the influence that different mitigation strategies have on system performance. Two strategies for preventing overcurrenting were presented, and their influence on operational performance was evaluated.

11.2 Human Factors in Crane Control

Part two of this thesis considered the role of the human operator in crane manipulation. This multifaceted function combines skilled manual control with supervisory tasks, such as monitoring throughput, assessing potential hazards, and payload transport decision making. For effective crane control, operators must possess a great deal of skill in each of these areas.

Fundamental theories of human factors were used to develop a model of human/crane interaction. The various tasks that operators perform were associated with the cognitive resource use and complexity required by each task. This model revealed that some tasks are more difficult, in terms of mental exertion, to perform than others. Specifically, the manual control task, in which operators attempt to manipulate the crane along a desired trajectory, is very difficult. It requires the two highest levels of cognitive exertion: rule-based and critical-thinking-based cognition.

A second model of human/crane interaction was developed that represents the human operator as a control element within a feedback loop. In this capacity, the operator is continuously comparing the crane state with a desired state, and then making adjustments to the crane trajectory as needed. This also requires rule-based and critical-thinking-based cognitions.

The models of human/crane interaction motivated the development of the visual touch-screen interface. This interface is fundamentally different from conventional interface devices. It permits operators to command crane motion using intuition-based behavior, rather than more difficult rule-based or critical-thinking-based behavior. It also removes the operator from the manual manipulation feedback loop, and places him or her entirely in a supervisory role. In this capacity, the operator has an increased awareness of the state, and can better manage safety and efficiency concerns.

To assess the operational effects of the visual interface, several operator studies were conducted where the performance of the visual interface was compared to that of conventional interfaces. The results of these studies, in conjunction with a trajectory and interface classification scheme, were used to demonstrate that the visual interface has several advantageous benefits over conventional interface devices.

11.3 Case Studies of Industrial Crane Control

Part three of this thesis demonstrated the pragmatic implementation of nonlinear input-shaping theory and interface design in a real-world industrial setting. A 30-ton and 35-ton bridge crane were equipped with a crane manipulation system (CMS). The CMS enabled operators to manipulate the cranes in a nearly swing-free manner, and also provided operators with a means to precisely position the cranes.

11.4 Research Extensions

Several opportunities for future research have arisen out of this work. The following presents some of the topics where dedicated efforts could contribute substantially to the areas of input shaping and human/machine interaction.

11.4.1 Stability of Variable Saturation and Rate Limiting Mitigation

An important objective of this work was to present mitigation strategies for negating the detrimental effects of actuator nonlinearities. On such technique was presented in Section 4.1.2. It involves actively conditioning input signals so that saturation or rate limiting nonlinearities in the controlled plant are not excited. Section 4.1.4 pointed out how this mitigation technique is actually a form of feedback control since past values of a signal are used in real time to adjust current signal values. Consequently, a method for conservatively implementing the mitigation technique was presented, which provides a pragmatic means for avoiding potential undesired dynamic behavior. While this technique was used successfully for the purpose it was designed, a more thorough understanding of the stability of this dynamic effect would be extremely useful. Because of the discontinuous behavior of the mitigation technique, obtaining a thorough understanding of variable saturation and rate limiting stability is both a challenging and intriguing prospect. This task in not trivial and will require dedicated effort.

11.4.2 Generalization of Nonlinear Input-Shaping Theory

Because of the ubiquitous presence of actuator nonlinearities, it is important for input-shaping practitioners to understand how these elements affect input-shaping control. Consequently, this work provided a very thorough investigation of these effects for three frequently used input-shaping filters: the ZV, ZVD, and UMZV shapers. While many of the conclusions resulting from this investigation are clearly extendable to a broader class of input shapers, it remains unclear whether or not other results can be applied generally. Until additional investigations are conducted, this will remain an open question.

11.4.3 Application Extension

The advancements made in the areas of nonlinear input-shaping theory and human/machine interface design were presented in the context of crane control. Cranes provide an excellent test bed for these developments because of their inherent dynamic properties and manipulation challenges. Nevertheless, the work presented here can benefit a much broader application space. For example, nuclear clean-up efforts currently underway at the Hanford Site in Washing State involve accessing large waste-storage tanks through small portals. This endeavor invariably involves long, slender, and flexible manipulation devices that are operated remotely. The advancements made in the areas of dynamic control and human/machine interaction may significantly improve the controllability and utility of these devices.

APPENDIX A

DERIVATION AND ALTERNATIVE EXPRESSIONS OF THE *R*-VALUE

Consider an arbitrary command, $x(t)$, acting on a linear second-order system, G, with a natural frequency, damped natural frequency, and damping ratio of ω_n, ω_d, and ζ, respectively.

The amplitude of residual oscillation induced into G by $x(t)$ can be determined by interpreting $x(t)$ as the product of a convolution between a unit step command, and an unknown function, $IS(t)$. To facilitate this development, two key assumptions about $x(t)$ are made, from which two key corollaries follow:

Assumption: $x(t) = 0$ for $t \leq 0$ (Assumption 1)

Assumption: $x(t) = C$ for $t \geq t_f$, $C \in \mathbb{R}$ (Assumption 2)

Corollary: $\dot{x}(t) = 0$ for $t < 0$ (Corollary 1)

Corollary: $\dot{x}(t) = 0$ for $t > t_f$ (Corollary 2)

A.1 Determination of the Unknown Function, IS(t)

In general, the convolution relationship between the unit-step command, the arbitrary signal, and the unknown function can be expressed in the Laplace domain as:

$$X(s) = \frac{1}{s} IS(s) \tag{A.1}$$

from which one obtains:

$$IS(s) = sX(s) \tag{A.2}$$

The derivative property of the Laplace transformation specifies that for a function, $f(t)$:

$$\mathcal{L}[\dot{f}(t)] = sF(s) - f(0) \tag{A.3}$$

When applied to $x(t)$, (A.3) yields:

$$\mathcal{L}[\dot{x}(t)] = sX(s) - x(0) \tag{A.4}$$

By Assumption 1, $x(0) = 0$, consequently, (A.4) can be substituted into (A.2) to yield:

$$IS(s) = \mathcal{L}[\dot{x}(t)] \tag{A.5}$$

from which one obtains:

$$IS(t) = \dot{x}(t) \tag{A.6}$$

The right-hand side of (A.6) can be expressed as a sequence of impulses by making use of the Dirac delta function, $\delta(t)$, yielding:

$$\dot{x}(t) = \int_{-\infty}^{\infty} \dot{x}(\tau)\delta(t-\tau)d\tau \tag{A.7}$$

However, by Corollary 1 and Corollary 2, the upper and lower integration bounds of (A.7) can be set to finite values, thus:

$$IS(t) = \dot{x}(t) = \int_{0^+}^{t_f} \dot{x}(\tau)\delta(t-\tau)d\tau \tag{A.8}$$

A.2 Phase Plane Representation of IS(t)

The sequence of impulses defined by (A.8) may be interpreted as an input shaper. Consequently, the vibratory properties of the input shaper can be assessed by considering each impulse from the perspective of the phase plane. Accordingly, each impulse is transformed into a vector contained in the phase plane by an appropriate scaling and angle assignment. Equations (2.5) and (2.4) define the correct scaling and angle terms, respectively. Applied to (A.8), these equations yield the vector representation of each impulse:

$$\dot{\boldsymbol{x}}(t) = \overbrace{e^{-\zeta\omega_n(t_f-t)}}^{\text{scaling term}}\,\dot{x}(t)\quad\overbrace{e^{j\omega_d t}}^{\text{angle term}} = \dot{x}(t)e^{-\zeta\omega_n t_f}e^{t(\zeta\omega_n+j\omega_d)} \tag{A.9}$$

A.3 Time Domain Representation of the Resultant Vector

The *resultant vector sum*, $\boldsymbol{R}$, of the impulse sequence provides a quantitative description of oscillation induced into G by $x(t)$. The magnitude of $\boldsymbol{R}$ is proportional to the amplitude

of residual oscillation exhibited by G at time $t = t_f$. More specifically, $|\boldsymbol{R}|$ is equal to the ratio of two measurable quantities:

$$|\boldsymbol{R}| = \frac{A_{x(t)}}{A_{step}} \tag{A.10}$$

where $A_{x(t)}$ is the amplitude of residual oscillation, measured at time $t = t_f$, induced into G by $x(t)$; and A_{step} is the amplitude of residual oscillation, measured at time $t = 0$, that would be exhibited by G in response to a unit step command. The angle of $\boldsymbol{R}$ is also physically significant. It is equal to the phase shift of the residual oscillation exhibited by G in response to $x(t)$.

To calculate $\boldsymbol{R}$, each vector defined by A.9 can be added vectorially to one another. This is accomplished by integration of A.9 as t is varied from 0^+ to t_f:

$$\boldsymbol{R} = e^{-\zeta\omega_n t_f} \int_{0^+}^{t_f} \dot{x}(t) e^{t(\zeta\omega_n + j\omega_d)} dt \tag{A.11}$$

A.4 Laplace Domain Representation of the Resultant Vector

Consider the Laplace representation of $\dot{x}(t)$:

$$\mathcal{L}[\dot{x}(t)] = \int_{-\infty}^{\infty} \dot{x}(t) e^{-st} dt \equiv sX(s) - x(0) \tag{A.12}$$

In light of Assumption 1, Corollary 1, and Corollary 2, (A.12) simplifies to:

$$\mathcal{L}[\dot{x}(t)] = \int_{0^+}^{t_f} \dot{x}(t) e^{-st} dt \equiv sX(s) \tag{A.13}$$

By evaluating (A.13) when $s = -\zeta\omega_n - j\omega_d$, one obtains:

$$[sX(s)]_{s=-\zeta\omega_n - j\omega_d} = \int_{0^+}^{t_f} \dot{x}(t) e^{t(\zeta\omega_n + j\omega_d)} dt \tag{A.14}$$

which is precisely the integral portion of (A.11). Therefore,

$$\boldsymbol{R} = e^{-\zeta\omega_n t_f} [sX(s)]_{s=-\zeta\omega_n - j\omega_d} \tag{A.15}$$

A.5 Digital Domain Representation of the Resultant Vector

The resultant vector, $\boldsymbol{R}$, is, in fact, a phaser that represents the residual oscillation of G. Accordingly, from time t_f onward, $\boldsymbol{R}$ attenuates in magnitude at the same rate that the

damping inherent to G attenuates the residual oscillation amplitude. Moreover, the phase angle of $\boldsymbol{R}$ also increases at a rate corresponding to the damped natural frequency of G. Consequently, the magnitude and phase of $\boldsymbol{R}$ after time t_f can be determined from:

$$\boldsymbol{R}\big|_{t>t_f} = \overbrace{e^{-\zeta\omega_n(t-t_f)}}^{\text{attenuating term}} \boldsymbol{R}\big|_{t=t_f} \overbrace{e^{j\omega_d(t-t_f)}}^{\text{angle term}} \tag{A.16}$$

which simplifies to:

$$\boldsymbol{R}\big|_{t>t_f} = \boldsymbol{R}\big|_{t=t_f} e^{(-\zeta\omega_n+j\omega_d)(t-t_f)} \tag{A.17}$$

Because (A.17) describes the behavior of $\boldsymbol{R}$ throughout a given time interval, it is useful for establishing equations that are well suited for calculating $\boldsymbol{R}$ when $x(t)$ is not known in closed form. The following sections provide insight into this useful application.

A.5.1 Calculation of $\boldsymbol{R}$ when $x(t)$ is Approximated with a Series of Steps

Suppose that $x(t)$ is approximated numerically with a series of step commands, as illustrated in Figure A.1.

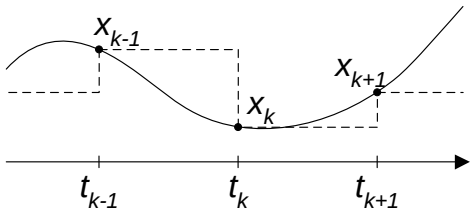


Figure A.1: Approximation of a Continuous Time Signal with Step Commands.

The $\boldsymbol{R}$-value representation of oscillation caused by the sampled command at time t_k can be denoted as $\boldsymbol{R}_k$. It is equal to the $\boldsymbol{R}$-value arising from the step at time t_k, added vectorially to $\boldsymbol{R}_{k-1}$, the $\boldsymbol{R}$-value representation of oscillation at time t_{k-1}. To compensate for the time interval between t_{k-1} and t_k, $\boldsymbol{R}_{k-1}$ is attenuated, and its phase is changed according to A.17. This leads to a simple digitally implementable representation of the $\boldsymbol{R}$-value when $x(t)$ is approximated with a series of step commands:

$$\boldsymbol{R}_k = \Delta x + \boldsymbol{R}_{k-1} e^{(-\zeta\omega_n+j\omega_d)\Delta t} \tag{A.18}$$

where

$$\Delta t = t_k - t_{k-1}$$

$$\Delta x = x_k - x_{k-1}$$

The term, Δx, is the $\boldsymbol{R}$-value arising from the step at time t_k.

A.5.2 Calculation of $\boldsymbol{R}$ when $x(t)$ is Approximated with a Series of Ramps

Approximating an arbitrary function with a series of step commands can lead to gross approximation error if the signal is coarsely sampled. Greater accuracy can be achieved if the number of samples is increased, or ramp commands are used to approximate the signal instead of step commands. Figure A.2 illustrates how a continuous time signal can be approximated by a series of ramp commands.

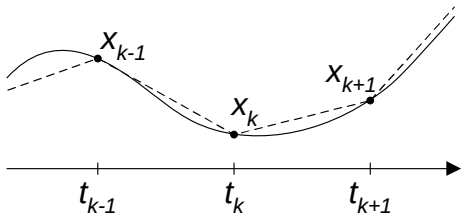


Figure A.2: Approximation of a Continuous Time Signal with Ramp Commands.

The $\boldsymbol{R}$-value representation of oscillation caused by the ramp-approximated command can be calculated in a similar manner as the step-approximated command. $\boldsymbol{R}_k$ is equal to the $\boldsymbol{R}$-value arising from the ramp terminating at time t_k, added vectorially to $\boldsymbol{R}_{k-1}$, the $\boldsymbol{R}$-value representation of oscillation at time t_{k-1}. The phase and amplitude of $\boldsymbol{R}_{k-1}$ is adjusted according to A.17. This leads to a simple digitally implementable representation of the $\boldsymbol{R}$-value when $x(t)$ is approximated with a series of ramp commands:

$$\boldsymbol{R}_k = \left\{ e^{-\zeta\omega_n \Delta t} \left[\frac{\Delta x/\Delta t}{s}(1 - e^{-s\Delta t}) \right]_{s=-\zeta\omega_n - j\omega_d} \right\} + \boldsymbol{R}_{k-1} e^{(-\zeta\omega_n + j\omega_d)\Delta t} \tag{A.19}$$

where Δx and Δt were defined previously. The term in curly brackets is the $\boldsymbol{R}$-value arising from the ramp terminating at time t_k.

APPENDIX B

INPUT/OUTPUT RELATIONSHIPS FOR COMMON ACTUATOR NONLINEARITIES

Several actuator nonlinearities have been addressed in this thesis. These include saturation, rate limiting, dead-zone, backlash, discretization, and switching. The following sections provide a comprehensive, yet succinct, compilation of the behavior of these nonlinearities. If they exist, the behavior of the inverse nonlinearity is also provided.

B.1 Saturation

The general relationship between the input and output of a saturator, $SAT(\cdot)$, with input x, is graphically illustrated in Figure B.1. The upper and lower saturation thresholds are p_U and p_L, respectively. Saturation can be mathematically described by:

$$SAT(x) = \begin{cases} p_U, & x > p_U \\ p_L, & x < p_L \\ x \ , & \text{otherwise} \end{cases} \tag{B.1}$$

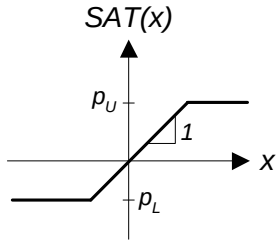


Figure B.1: Input/Output Relationship for a Saturator.

B.2 Rate Limiting

The functional relationship between the input and output of a rate limiting element, $RL(\cdot)$ with input x, can be described in the continuous time domain through the use of the elementary block diagram depicted in Figure B.2. The output of the rate limiter is fed back

in order to drive the error between the input and output to zero. This error is multiplied by an infinite gain, and subsequently saturated. The function of the saturator within this block diagram is to enforce that the upper and lower rate thresholds, m_U and m_L, respectively, are enforced.

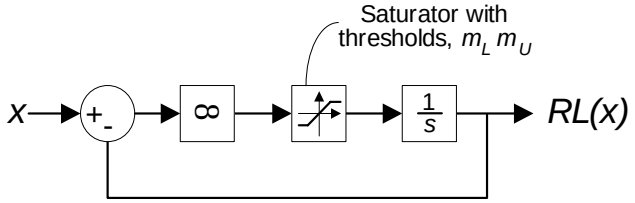


Figure B.2: Block Diagram Representation of Rate Limiting.

While Figure B.2 provides a conceptual basis for the function of a rate limiter, it is impractical for implementation purposes. Rate limiting is most easily implemented in the digital domain. A discretely sampled signal, $x(t_k) = x_k$, can be rate limited using the following rate limiting rules:

$$RL(x_k) = \begin{cases} RL(x_{k-1}) + m_U(t_k - t_{k-1}), & \frac{x_k - RL(x_{k-1})}{t_k - t_{k-1}} > m_U \\ RL(x_{k-1}) + m_L(t_k - t_{k-1}), & \frac{x_k - RL(x_{k-1})}{t_k - t_{k-1}} < m_L \\ x_k & , \text{ otherwise} \end{cases} \tag{B.2}$$

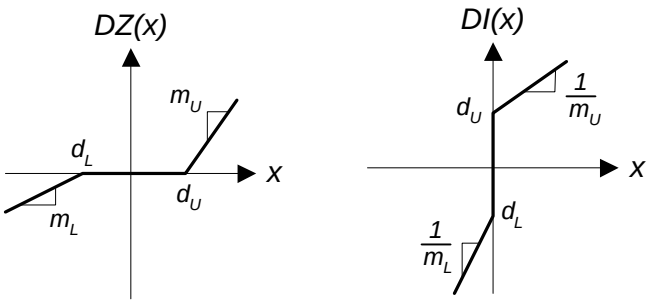
B.3 Dead-Zone and Inverse Dead-Zone[1]

The general relationship between the input and output of a dead-zone element, $DZ(\cdot)$, with input x, is graphically illustrated in Figure B.3(a). This relationship can be mathematically described by:

$$DZ(x) = \begin{cases} m_U(x - d_U), & x > d_U \\ 0 & , \ d_L \leq x \leq d_U \\ m_L(x - d_L), & x < d_L \end{cases} \tag{B.3}$$

The general relationship between the input and output of an inverse dead-zone element, $DI(\cdot)$, with input x, is graphically illustrated in Figure B.3(b). This relationship can be

[1]The contents of this section were adapted from [119].



(a) Dead-Zone. (b) Inverse Dead-Zone.

Figure B.3: Input/Output Relationship for a General Dead-Zone and Inverse Dead-Zone.

mathematically described by:

$$DI(x) = \begin{cases} \frac{1}{m_U}x + d_U), & x > 0 \\ \frac{1}{m_L}x + d_L), & x < 0 \\ 0 \qquad\qquad , & x = 0 \end{cases} \tag{B.4}$$

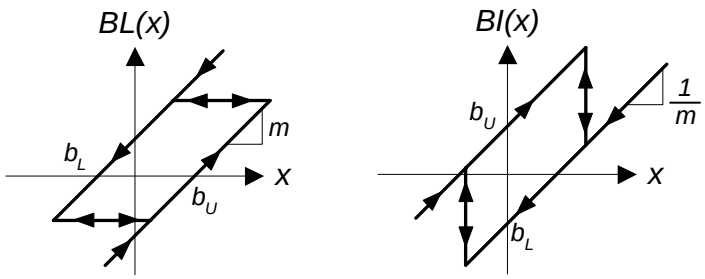
B.4 Backlash and Inverse Backlash[2]

Backlash behavior, $BL(\cdot)$, with input x, is described by two parallel functional lines: the upward and downward sides of $BL(\cdot)$. This is illustrated in Figure B.4(a). Transition from the upward function to the downward function, or vice versa, is achieved across a horizontal function. The precise behavior of the backlash element can be concisely described in the domain of $\dot{BL}(\cdot)$ and $\dot{x}$:

$$\dot{BL}(x) = \begin{cases} m\dot{x}, & \dot{x} > 0 \text{ and } BL(x) = m(x - b_U) \\ m\dot{x}, & \dot{x} < 0 \text{ and } BL(x) = m(x - b_L) \\ 0 \quad , & \text{otherwise} \end{cases} \tag{B.5}$$

Inverse backlash behavior, $BI(\cdot)$, with input x, is graphically illustrated in Figure B.4(b). Like the mathematical description for backlash, inverse backlash can be concisely described

[2]The contents of this section were adapted from [118].



(a) Backlash. (b) Inverse Backlash.

Figure B.4: Input/Output Relationship for a General Backlash and Inverse Backlash.

in the domain of its derivative:

$$\dot{BI}(x) = \begin{cases} \frac{1}{m}\dot{x} & , \ \dot{x} > 0 \text{ and } BI(x) = \frac{x}{m} + b_U \\ \frac{1}{m}\dot{x} & , \ \dot{x} < 0 \text{ and } BI(x) = \frac{x}{m} + b_L \\ 0 & , \ \dot{x} = 0 \\ \delta(0)(b_U - b_L) & , \ \dot{x} > 0 \text{ and } BI(x) = \frac{x}{m} + b_L \\ -\delta(0)(b_U - b_L), & \dot{x} < 0 \text{ and } BI(x) = \frac{x}{m} + b_U \end{cases} \tag{B.6}$$

B.5 Discretization

The function of discretization, $D(\cdot)$, is to round the values of a given input, x, to the nearest admissible state. The functional relationship between the input and output of a general four-state discretizer is shown in Figure B.5.

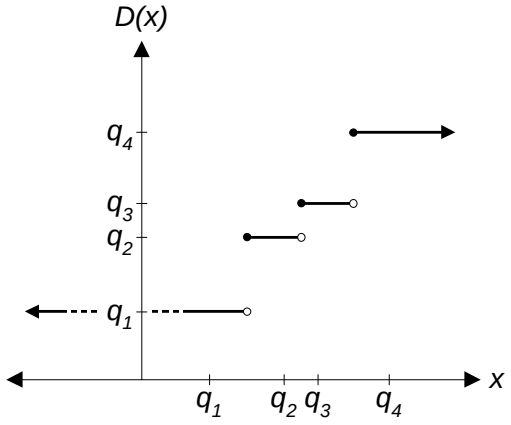


Figure B.5: Input/Output Relationship for a General Four-State Discretizer.

The behavior of a general m-state discretizer with admissible states $\mathbf{Q} = \{q_1, q_2, \ldots, q_m\}$,

where $q_k < q_{k+1}$, can be expressed mathematically as:

$$D(x) = \begin{cases} q_1 \ , \ x < \frac{1}{2}(q_1 + q_2) \\ q_2 \ , \ \frac{1}{2}(q_1 + q_2) \leq x < \frac{1}{2}(q_2 + q_3) \\ \vdots \ , \quad \vdots \\ q_k \ , \ \frac{1}{2}(q_{k-1} + q_k) \leq x < \frac{1}{2}(q_k + q_{k+1}) \\ \vdots \ , \quad \vdots \\ q_m, \ q_m \leq x \end{cases} \tag{B.7}$$

B.6 Switching on AC Induction Motors and Vector Drives

The behavior of AC induction motors and vector drives is nonlinear. However, in the velocity domain, this behavior can be accurately modeled by combining several simpler components [111]. The DM block of Figure B.6 reveals such a model.

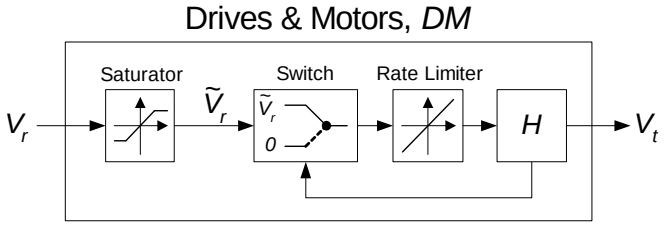


Figure B.6: Block Diagram Model of AC Induction Motors and Vector Drives.

The model is comprised of four elements: a saturator, a switch, a rate limiter, and a heavily damped second-order plant, H. The saturation element truncates excessive velocity commands to the crane, while the rate limiter places upper and lower bounds on the acceleration of the crane. H serves to mimic the smoothing behavior of the drives and motors. The function of the switching element is to pass the reference velocity signal, $\widetilde{V}_r$, to the rate-limiting block. However, when transitional velocity commands are issued to the crane, the switch temporarily sends a signal of zero. Transitional velocity commands are those commands that change the direction of travel of the crane (forward to reverse or vise versa). This type of behavior depends on $\widetilde{V}_r$ and the velocity output of the drives and motors, V_t.

It can be described with the following switching rules:

$$\text{Switch Output} = \begin{cases} \widetilde{V}_r, \ \text{Sign}(\widetilde{V}_r) = \text{Sign}(V_t) \\ \widetilde{V}_r, \ |V_t| \leq X \\ 0 \ , \ \text{otherwise} \end{cases} \tag{B.8}$$

To mimic the behavior of a given set of drives and motors, one must properly select the parameters associated with the model: the saturation thresholds, X - the switching threshold, the slopes of the rate limiter, ζ_H - the damping ratio of H, and ω_{nH} - the natural frequency of H.

APPENDIX C

TESTING INFINITE-ACCELERATION COMMANDS ON FINITE-ACCELERATION MACHINES[1]

Chapter 3 dealt extensively with quantifying the oscillatory effects of various actuator nonlinearities on input-shaped control of flexible machines. A challenging aspect of this endeavor is related to experimental validation. This is due to the fact that on physical systems, it can be difficult to isolate the oscillation caused by a single nonlinearity from the oscillation caused by other nonlinearities. For example, the oscillatory effects of dead-zone cannot be completely decoupled from those arising from rate limiting. This is because rate limiting is an inherent behavior of every physical plant. The following sections address this specific difficulty for the case of ZV input-shaping control.

C.1 Measuring the Effects of Dead-Zone on ZV Input Shapers

Consider the 10-ton bridge crane described previously in Chapter 3. A serial block diagram model of this crane was presented where the nonlinear behavior of the drives and motors was captured in a block, NP, and the linear relationship between the crane velocity and the payload swing was captured in the block, G.

The results of Section 3.2.1 predict that if ZV-shaped step commands are issued to the crane, and significant dead-zone behavior exists, then the shaped commands will be corrupted, thus causing the payload to oscillate. These predictions were previously summarized in the $\boldsymbol{R}$-value plot of Figure 3.18. A method to experimentally validate these results on a system *without* rate limiting consists of the following four steps:

1. Issue a velocity step command of magnitude H to the system with the dead-zone width, d, set to zero. Measure the magnitude of the residual payload oscillation in response to this command. This measurement is equal to $H \cdot P_{unit}$, where P_{unit} is

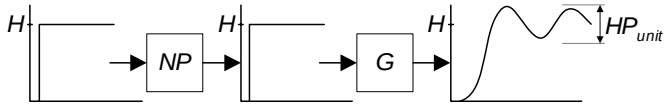
[1] Portions of this appendix are published in [101, 102].

the amplitude of oscillation induced by a unit step command. This measurement is illustrated in Figure C.1(a).

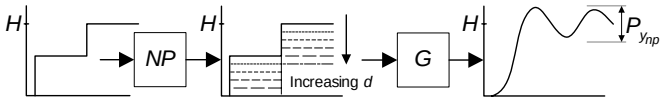
2. Similarly, issue ZV-shaped velocity step commands to the system. Measure $P_{y_{np}}$ for different values of d. $P_{y_{np}}$ is the amplitude of oscillation exhibited in response to the shaped commands. This measurement is illustrated in Figure C.1(b).

3. Normalize the $P_{y_{np}}$ values obtained in Step 2 by $H \cdot P_{unit}$. According to (A.10), the resulting ratios, $P_{y_{np}}/(H \cdot P_{unit})$, are equal to $|\boldsymbol{R}|/H$. They represent the experimentally obtained oscillatory effects of dead-zone on ZV-shaped step commands.

4. Compare the measured values of $|\boldsymbol{R}|/H$ with the theoretical predictions of Figure 3.18.



(a) Oscillation Induced by a Step Command.

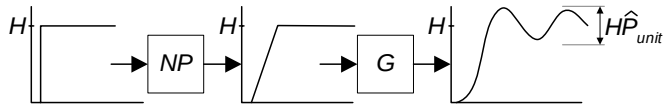


(b) Oscillation Induced by a Dead-Zone Affected Staircase Command.

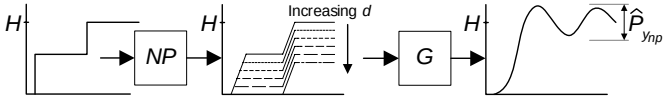
Figure C.1: P-Values for a System with Dead-Zone.

One attribute of this procedure is important to consider: physical systems are incapable of instantaneously changing velocity. Therefore, the drives and motors cannot move the crane with the infinite acceleration commands shown in Figure C.1. Instead, velocity profiles will be issued to G that are consistent with the maximum achievable acceleration (rate limit) of the system. This scenario is illustrated in Figure C.2. The slope of the velocity profiles is indicative of the rate limit inherent to the drives and motors plant, NP.

Because both rate limiting and dead-zone are present in the system, the compound effects of these nonlinearities will be exhibited in the measurements of residual oscillation. Therefore, the amplitudes P_{unit} and $P_{y_{np}}$ that would be obtained in the case that infinite acceleration were possible will be replaced with $\widehat{P}_{unit}$ and $\widehat{P}_{y_{np}}$. These amplitudes are shown



(a) Oscillation Induced by a Rate Limited Step Command.



(b) Oscillation Induced by a Dead-Zone and Rate Limited Affected Staircase Command.

Figure C.2: $\widehat{P}$-Values for a System with Dead-Zone and Rate Limiting.

in Figure C.2. A natural question arising from this circumstance is: *Can measurements of* $\widehat{P}_{unit}$ *and* $\widehat{P}_{y_{np}}$ *be used instead of* P_{unit} *and* $P_{y_{np}}$ *to validate the predicted effects of dead-zone*?

C.2 Isolating the Effects of Dead-Zone from Rate Limiting

To address the posed question, a ratio of $\widehat{P}$-values can be evaluated as the severity of the rate-limit is varied. The desired ratio can be obtained through an $\boldsymbol{R}$-value analysis because of the following equality:

$$\frac{\widehat{P}_{y_{np}}}{H \cdot \widehat{P}_{unit}} = \frac{|\widehat{\boldsymbol{R}}_{y_{np}}|}{|\widehat{\boldsymbol{R}}_{unit}|} \tag{C.1}$$

$\widehat{\boldsymbol{R}}_{y_{np}}$ are the $\boldsymbol{R}$-values for the rate-limited/dead-zone signals previously shown in Figure C.2(b), and $\widehat{\boldsymbol{R}}_{unit}$ are the $\boldsymbol{R}$-values for the rate-limited step command shown in Figure C.2(a).

C.2.1 *R*-Value Analysis:

The $\boldsymbol{R}$-value for an arbitrary command, $y_{np}(t)$, acting on an undamped harmonic oscillator may be determined in the Laplace domain from:

$$|\boldsymbol{R}| = |sY_{np}(s)|_{s=-j\omega} \tag{C.2}$$

Applying (C.2) to the rate-limited step command of Figure C.2(a) produces:

$$|\widehat{\boldsymbol{R}}_{unit}| = \left| \frac{KH}{2\pi j} \left(1 - e^{2\pi j/K}\right) \right| \tag{C.3}$$

where

$$K = \frac{T}{t_r} \tag{C.4}$$

K is a non-dimensional ratio that indicates the severity of the acceleration limit. It is the ratio between the period of oscillation for the harmonic system, and the rise time of the acceleration-limited step command. A large value of K indicates that the system is capable of achieving high accelerations. A small value of K indicates that the system is sluggish.

Applying (C.2) to the rate limited, staircase commands of Figure C.2(b) produces:

$$|\widehat{\boldsymbol{R}}_{y_{np}}| = \begin{cases} \left| \frac{KH}{2\pi j} \left[e^{jt_a} + e^{jt_b} \right] \right|, & 0 \le d < \frac{H}{2} \\ \left| \frac{KH}{2\pi j} \left[1 + e^{jt_c} \right] \right| & , \ \frac{H}{2} \le d \le H \end{cases} \tag{C.5}$$

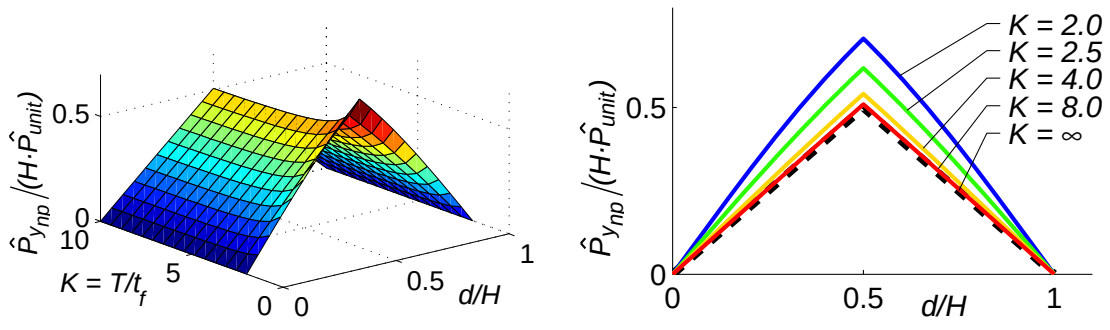
where

$$t_a = \frac{2\pi}{K} \left(\frac{1}{2} - \frac{d}{H} \right) \tag{C.6}$$

$$t_b = \frac{\pi}{K} (1 + K) \tag{C.7}$$

$$t_c = \frac{\pi}{K} (2 - 2d + K) \tag{C.8}$$

The required ratio is obtained by dividing (C.5) by (C.3). The result is a scalar function of the dead-zone parameter, d, and the parameter describing the severity of the rate limit, K. The value of this ratio forms the surface shown in Figure C.3(a). Sections of the surface are shown in Figure C.3(b) for various K values ranging from $K = 2$ to $K = \infty$.



(a) Surface Plot.

(b) Section Plot.

Figure C.3: Oscillatory Effects of Acceleration-Limited, Dead-Zone Commands.

The case when $K = \infty$ is significant. This value of K results when the physical system is not restricted by a rate limit. Thus, the line corresponding to $K = \infty$ represents the oscillatory effects of pure dead-zone on ZV-shaped step commands, without the influence of a rate-limit. Notice that for values of $K \geq 4$, the corresponding lines are nearly indistinguishable from one another. The 10-ton crane has a nominal rate limit corresponding to a K-value of approximately 8. For the experiments conducted in Sections 3.2.4 and 3.3.4, the rate limit of the crane was set at its maximum value of $28m_c$. This corresponds to a K-value of 28.

This result indicates that the inherent acceleration limit of the crane is negligible, and that:

$$\frac{\widehat{P}_{y_{np}}}{H \cdot \widehat{P}_{unit}} \cong \frac{P_{y_{np}}}{H \cdot P_{unit}} \equiv \frac{|\boldsymbol{R}|}{H}, \qquad \text{for } K \geq 4 \tag{C.9}$$

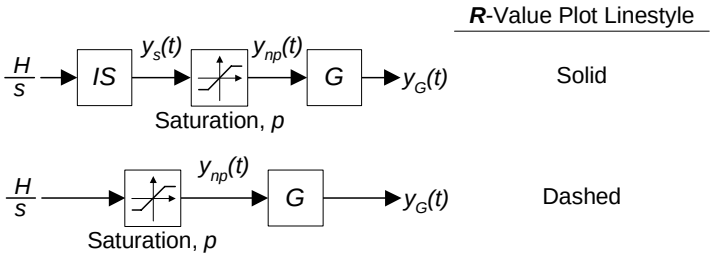
Therefore, given the light damping of the crane, the experimental procedure delineated in Section C.1 can be used confidently to measure the oscillatory effects of dead-zone on ZV-shaped step commands.

APPENDIX D

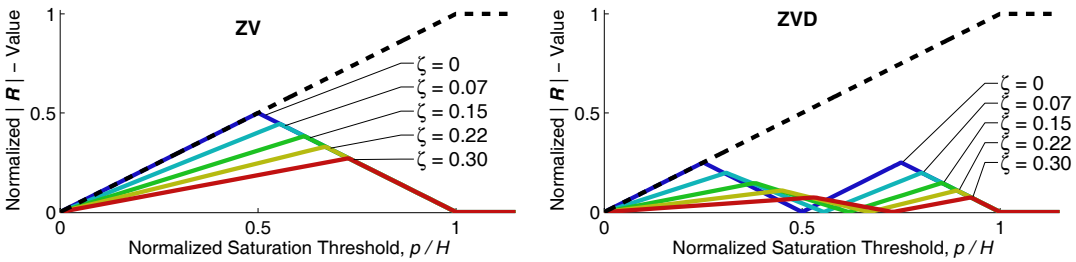
R-VALUE ANALYSIS SUMMARY FOR SATURATION, RATE LIMITING, DEAD-ZONE, & BACKLASH

D.1 Saturation

SYSTEMS

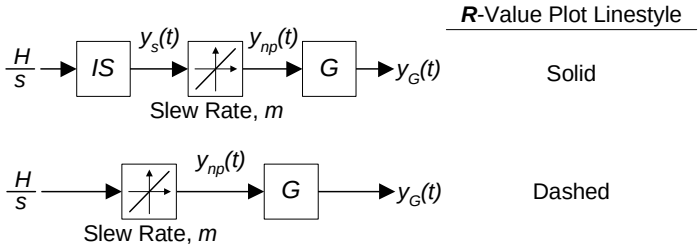


- G = second-order harmonic oscillator.
- Upper and lower saturation threshold specified by $\pm p$.
- Vertical axis value 1 $\Longrightarrow$ oscillation amplitude caused by step, magnitude H.

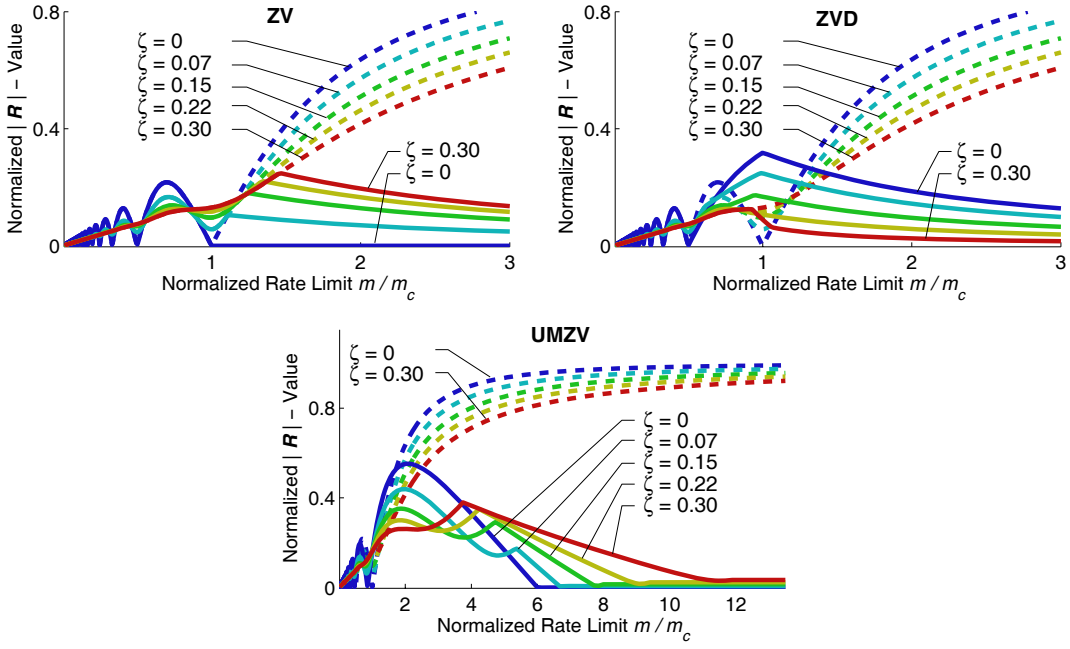


D.2 Rate Limit

SYSTEMS

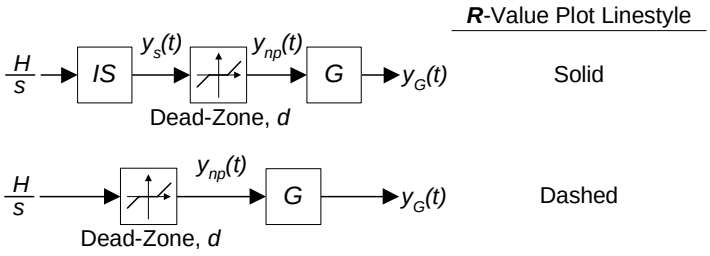


- G = second-order harmonic oscillator.
- Upper and lower rate threshold specified by $\pm m$.
- $m_c = H/T$. (T = damped period of oscillation).
- Vertical axis value 1 $\Longrightarrow$ oscillation amplitude caused by step, magnitude H.



D.3 Dead-Zone & Backlash

SYSTEMS

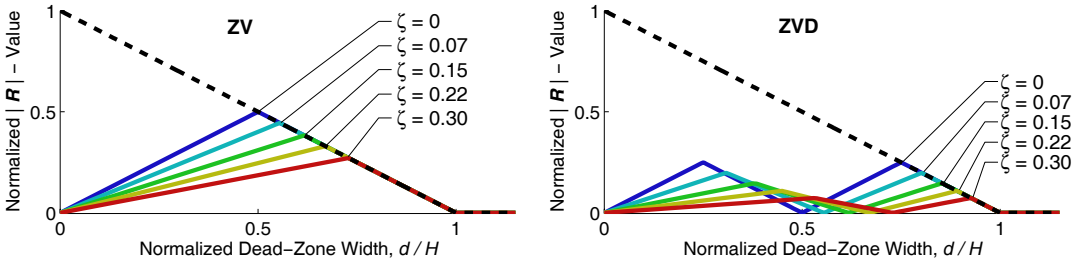


• G = second-order harmonic oscillator.

• Dead-zone width specified by $\pm d$.

• Vertical axis value 1 $\Longrightarrow$ oscillation amplitude caused by step, magnitude H.

• $\boldsymbol{R}$-value plots for backlash identical to $\boldsymbol{R}$-value plots for dead-zone, with dead-zone width, d, replace by backlash width, b.

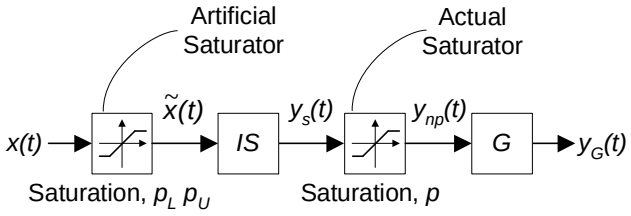


APPENDIX E

MITIGATION TECHNIQUE SUMMARY FOR SATURATION, RATE LIMITING, DEAD-ZONE, & BACKLASH

E.1 Saturation

MITIGATED SYSTEM



- G = second-order harmonic oscillator.
- Upper and lower threshold of actual saturator specified by $\pm p$.
- Upper and lower threshold of artificial saturator specified by p_U and p_L, respectively.

Fixed Saturation Constraints:

$$p_U = -p_L = \frac{p}{\sum_{i=1}^{n} |A_i|}$$

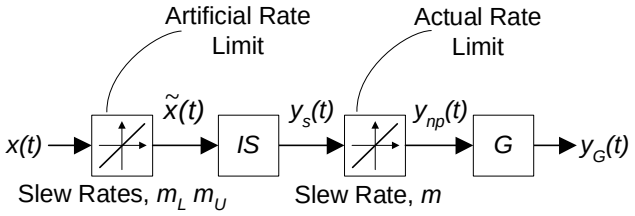
Variable Saturation Constraints:

$$p_L = \frac{1}{A_1}\left[-p - \sum_{i=2}^{n} A_i \widetilde{x}(t - t_i)\right]$$

$$p_U = \frac{1}{A_1}\left[p - \sum_{i=2}^{n} A_i \widetilde{x}(t - t_i)\right]$$

E.2 Rate Limit

MITIGATED SYSTEM



- G = second-order harmonic oscillator.
- Upper and lower threshold of actual rate limit specified by $\pm m$.
- Upper and lower threshold of artificial rete limit specified by m_U and m_L, respectively.

Fixed Rate Limit Constraints:

$$m_U = -m_L = \frac{m}{\sum_{i=1}^{n} |A_i|}$$

Variable Saturation Constraints:

$$s_1 = \frac{1}{A_1}\left[-m - \sum_{i=2}^{n} A_i \dot{\tilde{x}}(t - t_i)\right] \qquad s_2 = \frac{1}{A_1}\left[m - \sum_{i=2}^{n} A_i \dot{\tilde{x}}(t - t_i)\right]$$

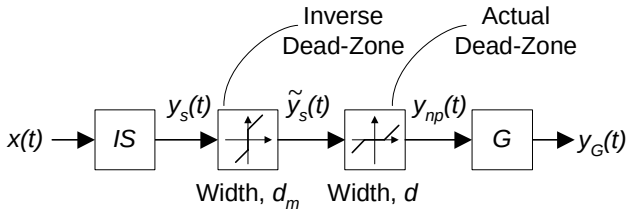
$$m_L = \begin{cases} s_1\alpha & , \ s_1 \leq 0 \\ s_1(2-\alpha), & s_1 > 0 \end{cases} \qquad m_U = \begin{cases} s_2\alpha & , \ s_2 \geq 0 \\ s_2(2-\alpha), & s_2 < 0 \end{cases}$$

Conservative Scaling Factor:

$$0.5 \lesssim \alpha \leq 1$$

E.3 Dead-Zone & Backlash

MITIGATED SYSTEM



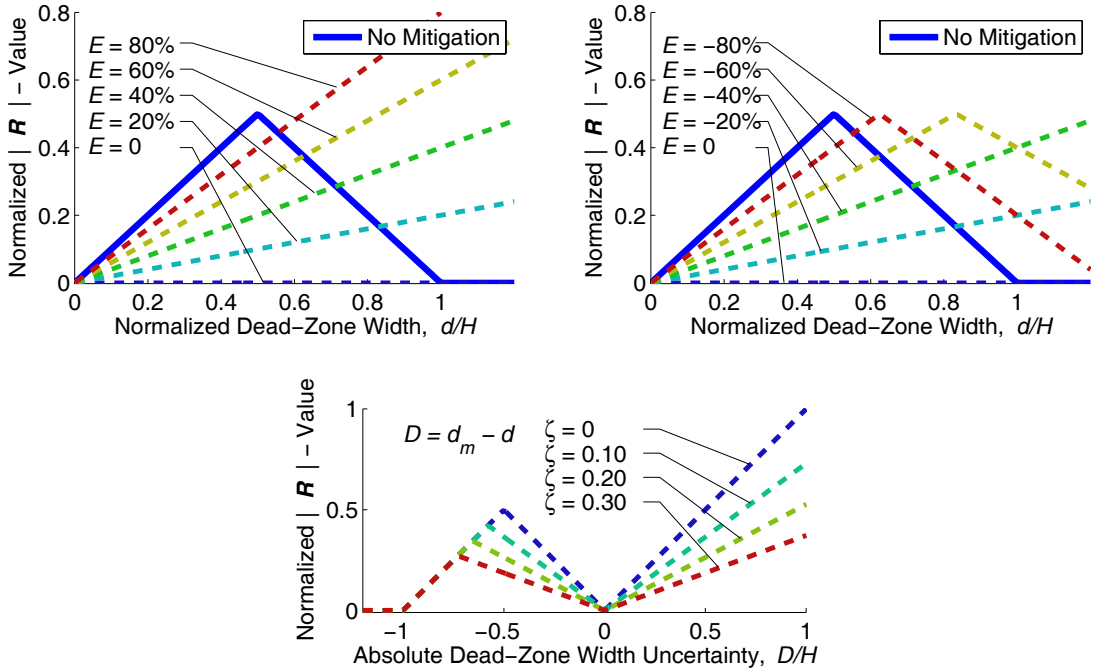
• G = second-order harmonic oscillator.

• Dead-Zone width specified by by $\pm d$.

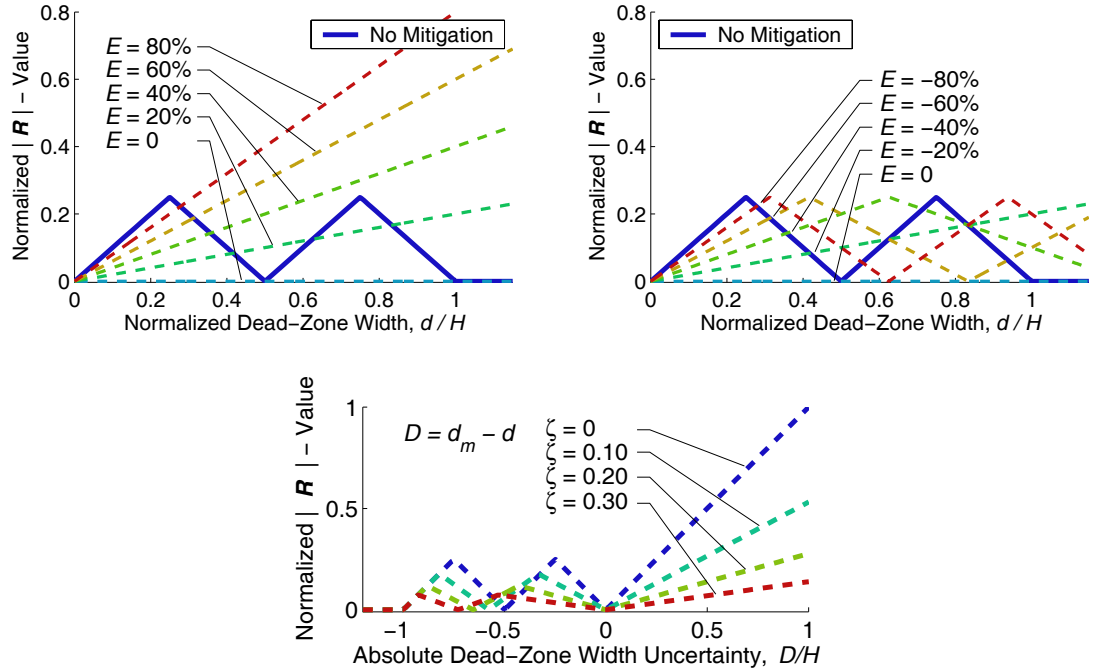
• Modeled dead-zone width specified by $\pm d_m$.

• Percent modeling error, $E \equiv (d_m - d)/d$. • Absolute modeling error, $D \equiv d_m - d$. • Mitigation technique and robustness curves for backlash identical to those for dead-zone, with dead-zone width and modeled dead-zone width, d and d_m, replace by backlash width and modeled backlash width, b and b_m, respectively.

MITIGATION ROBUSTNESS CURVES (ZV Shaper)



MITIGATION ROBUSTNESS CURVES (ZVD Shaper)



REFERENCES

[1] Agostini, M. J., Parker, G. G., Schaub, H., Groom, K., and Rush D. Robinett, I., "Generating swing-suppressed maneuvers for crane systems with rate saturation," *IEEE Trans. on Control Systems Technology*, vol. 11, no. 4, 2003.

[2] Alsop, C. F., Forster, G. A., and Holmes, F. R., "Ore unloader automation - a feasibility study," in *Proc. of the Tokyo Symposium on Systems Engineering for Control System Design*, (Tokyo), pp. 295 – 305, 1965.

[3] Amat, J., Frigola, M., and Casals, A., "Human robot interaction from visual perception," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (Sendal, Japan), pp. 1997–2002, 2004.

[4] Amato, F., Iervolino, R., Pandit, M., Scala, S., and Verde, L., "Analysis of pilot-in-the-loop oscillations due to position and rate saturations," in *Conf. on Decision and Control*, (Sydney, Australia), pp. 3564–3569, 2000.

[5] Andersch, P., Sorensen, K., and Singhose, W. E., "Effects of rate limiting on common input shaping filters," in *WSEAS Int. Conf. on Automatic Control, Modelling and Simulation*, (Istanbul, Turkey), 2008.

[6] Andersch, P., Sorensen, K., and Singhose, W. E., "Zv, zvd, and umzv input shaping on systems with rate limiting," *WSEAS Trans. on Systems and Control*, Accepted.

[7] Bhat, S. P. and Miu, D. K., "Precise point-to-point positioning control of flexible structures," *J. of Dynamic Systems, Measurement, and Control*, vol. 112, no. 4, pp. 667–674, 1990.

[8] Biediger, E. and Singhose, W., "Formation flying satellite control utilizing input shaping," in *AIAA/AAS Astrodynamics Specialist Conf.*, (Big Sky, Montana), 2003.

[9] Blackburn, D. F., Singhose, W., Kitchen, J. P., Petrangenaru, V. P., Lawrence, J., Kamoi, T., and Taura, A., "Advanced input shaping algorithm for nonlinear tower crane dynamics," in *8th Int. Conf. on Motion and Vibration Control*, (Daejeon, Korea), 2006.

[10] Bodson, M., "An adaptive algorithm for the tuning of two input shaping methods," *Automatica*, vol. 34, no. 6, pp. 771–776, 1998.

[11] Calvert, J., Wilmette, and Gimpel, D., "Method and apparatus for control of system output in response to system input," 1957.

[12] Chen, C.-T., *Linear System Theory and Design*. New York: Oxford University Press, 3rd ed., 1999.

[13] Cutforth, C. F. and Pao, L. Y., "Control using equal length shaped commands to reduce vibration," *IEEE Trans. on Control Systems Technology*, vol. 11, no. 1, 2003.

[14] Danielson, J., Lawrence, J., and Singhose, W., "Command shaping for flexible systems subject to constant acceleration limits," *J. of Dynamic Systems, Measurement, and Control*, In Press.

[15] Drapeau, V. and Wang, D., "Verification of a closed-loop shaped-input controller for a five-bar-linkage manipulator," in *IEEE Conf. on Robotics and Automation*, (Altanta, GA), pp. 216–221, 1993.

[16] Eloundou, R. and Singhose, W., "Interpretation of smooth reference commands as input-shaped functions," in *American Control Conf.*, (Anchorage, AK), pp. 4948–4953, 2002.

[17] Fang, Y., Dixon, W., Dawson, D., and Zergeroglu, E., "Nonlinear coupling control laws for a 3-dof overhead crane system," in *IEEE Conf. on Decision and Control*, (Orlando, Florida, USA), pp. 3766–3771, 2001.

[18] Fliess, M., "Automatique et corps differentiels," *Forum Mathematicum*, vol. 1, pp. 227–238, 1989.

[19] Fliess, M., Levine, J., and Rouchon, P., "A simplified approach of crane control via a generalized state-space model," in *IEEE Conf. on Decision and Control*, (Brighton, England), pp. 736–741, 1991.

[20] Fortgang, J., Singhose, W., Marquez, J. d. J., and Perez, J., "Command shaping for micro-mills and cnc controllers," in *American Control Conf.*, (Portland, OR), pp. 4531–4536, 2005.

[21] Frigola, M., Fernandez, J., and Aranda, J., "Visual human machine interface by gestures," in *IEEE Int. Conf. on Robotics and Automation*, (Taipei, Taiwan), pp. 386–391, 2003.

[22] Fujita, M., Kamata, M., and Miyata, K., "Clarification of cognitive skill in mechanical work and its application," *Int. J. of Human-Computer Interaction*, vol. 18, no. 1, pp. 105–124, 2005.

[23] Gorinevsky, D. and Vukovich, G., "Nonlinear input shaping control of flexible spacecraft reorientation maneuver," *J. of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 264–270, 1998.

[24] Grosser, K. and Singhose, W., "Command generation for reducing perceived lag in flexible telerobotic arms," *JSME Int. J.*, vol. 43, no. 3, pp. 755–761, 2000.

[25] Gurleyuk, S. S., "Optimal unity-magnitude input shaper duration analysis," *Archive of Applied Mechanics*, vol. 77, no. 1, pp. 63 – 71, 2007.

[26] Gustafsson, T. and Heidenback, C., "Automatic control of unmanned cranes at the pasir panjang terminal," in *IEEE Int. Conf. on Control Applications*, (Glasgow, Scotland, U.K.), pp. 180–186, 2002.

[27] Hekman, K. A. and Singhose, W. E., "A feedback control system for suppressing crane oscillation with on-off motors," *Int. J. of Control, Automation, and Systems*, vol. 5, pp. 223–233, 2007.

[28] Hollnagel, E., *Cognitive Reliability and Error Analysis Method: CREAM*. Oxford: Alden Group, 1998.

[29] Hong, K.-T., Huh, C.-D., and Hong, K.-S., "Command shaping control for limiting the transient sway angle of crane systems," *Int. J. of Control, Automation, and Systems*, vol. 1, no. 1, 2003.

[30] Huey, J. R. and Singhose, W., "Experimental verification of stability analysis of closed-loop input shaping controllers,," in *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, (Monterey, CA, 2005), 2005.

[31] Huey, J., *The Intelligent Combination of Input Shaping and PID Feedback Control*. PhD thesis, Georgia Institute of Technology, 2006.

[32] Huey, J. R., Sorensen, K. L., and Singhose, W. E., "Useful applications of closed-loop signal shaping controllers," *Control Engineering Practice*, vol. 16, no. 7, pp. 836–846, 2008.

[33] Hyde, J. M. and Seering, W. P., "Using input command pre-shaping to suppress multiple mode vibration," in *IEEE Int. Conf. on Robotics and Automation*, vol. 3, (Sacramento, CA), pp. 2604–2609, IEEE, 1991.

[34] Jang, J. O., Lee, P. G., Chung, H. T., and Jeon, G. J., "Output backlash compensation of systems using fuzzy logic," in *American Control Conf.*, (Denver, CO), pp. 2489 – 2490, 2003.

[35] Jones, S. and Ulsoy, A. G., "An approach to control input shaping with application to coordinate measuring machines," *J. of Dynamic Systems, Measurement, and Control*, vol. 121, no. June, pp. 242–247, 1999.

[36] Kapila, V., Tzes, A., and Yan, Q., "Closed-loop input shaping for flexible structures using time-delay control," in *38th Conf. on Decision and Control*, (Phoenix, AZ), pp. 1531–1566, 1999.

[37] Kapila, V., Tzes, A., and Yan, Q., "Closed-loop input shaping for flexible structures using time-delay control," *J. of Dynamic Systems, Measurement, and Control*, vol. 122, pp. 454–460, 2000.

[38] Kavanagh, R. C., "Improved quantization error characteristics in fir differentiators using irregular sampling," *IEEE Signal Processing Letters*, vol. 9, no. 10, pp. 326–328, 2002.

[39] Kenison, M. and Singhose, W., "Concurrent design of input shaping and proportional plus derivative feedback control," *J. of Dynamic Systems, Measurement, and Control*, no. September, 2002.

[40] Khalid, A., Huey, J., Singhose, W., Lawrence, J., and Frakes, D., "Human operator performance testing using an input-shaped bridge crane," *J. of Dynamic Systems, Measurement, and Control*, vol. 128, no. 4, pp. 835–841, 2006.

[41] Khalid, A., Singhose, W., Huey, J., and Lawrence, J., "Study of operator behavior, learning, and performance using an input-shaped bridge crane," in *Conf. on Control Applications*, (Taipei, Taiwan), 2004.

[42] Khorrami, F., Jain, S., and Tzes, A., "Adaptive nonlinear control and input preshaping for flexible-link manipulators," in *American Control Conf.*, vol. 3, (San Francisco, CA), pp. 2705–2709, ACC, 1993.

[43] Kim, D. and Singhose, E. W., "Human operator learning on double-pendulum bridge cranes," in *ASME Int. Mechanical Engineering Congress and Exposition*, (Seattle, WA, USA), 2007.

[44] Kim, Y.-S., Hong, K.-S., and Sul, S.-K., "Anti-sway control of container cranes: Inclinometer, observer, and state feedback," *Int. J. of Control, Automation, and Systems*, vol. 2, no. 4, pp. 435–449, 2004.

[45] Kinceler, R. and Meckl, P. H., "Input shaping for nonlinear systems," in *American Control Conf.*, (Seattle, Washington), pp. 914 –918, 1995.

[46] Kojima, H. and Singhose, W., "Adaptive deflection limiting control for slewing flexible space structures," *J. of Guidance, Control, and Dynamics*, vol. 30, pp. 61–67, 2007.

[47] Lawrence, J., Falkenberg, M., and Singhose, W., "Input shaping for a flexible, nonlinear, one-link robotic arm with backlash," in *Japan-USA Symposium on Flexible Automation*, (Denver, Colorado), 2004.

[48] Lawrence, J., Singhose, W., and Hekman, K., "Friction-compensating command shaping for vibration reduction," *J. of Vibration and Acoustics*, vol. 127, pp. 307–314, 2005.

[49] Liu, K.-P., You, W., and Li, Y.-C., "Combining a feedback linearization approach with input shaping for flexible manipulator control," in *Int. Conf. on Machine Learning and Cybernetics*, (Xi'an), pp. 561–565, 2003.

[50] Liu, Q. and Wie, B., "Robust time-optimal control of uncertain flexible spacecraft," *J. of Guidance, Control, and Dynamics*, vol. 5, no. 3, pp. 597–604, 1992.

[51] Love, L. J., Magee, D. P., and Book, W. J., "A comparison of joint control algorithms for teleoperated pick and place tasks using a flexible manipulator," in *IEEE Int. Conf. on Systems, Man, and Cybernetics*, vol. 2, pp. 1257–62, 1994.

[52] Magee, D. and Book, W., "The application of input shaping to a system with varying parameters," in *Japan/USA Symp. on Flexible Automation*, pp. pp. 519–25, 1992.

[53] Magee, D. P. and Book, W. J., "Filtering schilling manipulator commands to prevent flexible structure vibration," in *American Control Conf.*, (Baltimore, MD), pp. 2538–42, 1994.

[54] Magee, D. P. and Book, W. J., "Filtering micro-manipulator wrist commands to prevent flexible base motion," in *American Control Conf.*, (Seattle, WA), pp. 924–928, 1995.

[55] Manning, R. C., *Improving the Control of Two-Mode Flexible Systems with Input Shaping*. Masters, Georgia Institute of Technology, 2008.

[56] Moustafa, K. A. F., "Reference trajectory tracking of overhead cranes," *J. of Dynamic Systems, Measurement, and Control*, vol. 123, pp. 139–141, 2001.

[57] Muenchhof, M. and Singh, T., "Concurrent feedforward/feedback controller design using time-delay filters," in *AIAA Guidance, Navigation, and Control Conf.*, (Monterey, CA), 2002.

[58] O'Brien, J. F., McInroy, J. E., Bodtke, D., Bruch, M., and Hamann, J. C., "Lessons learned in nonlinear systems and flexible robots through experiments on a 6 legged platform," in *American Control Conf.*, (Philadelphia, Pennsylvania), 1998.

[59] O'Connor, W. J., "A gantry crane problem solved," *J. of Dynamic Systems, Measurement, and Control*, vol. 125, pp. 569–576, 2003.

[60] Pao, L. Y., Chang, T. N., and Hou, E., "Input shaper designs for minimizing the expected level of residual vibration in flexible structures," in *American Control Conf.*, (Albuquerque, NM), pp. 3542–3546, 1997.

[61] Pao, L. Y., "An analysis of the frequency, damping, and total insensitivities of input shaping designs," *J. of Guidance, Control, and Dynamics*, vol. 20, no. 5, pp. 909–915, 1997.

[62] Pao, L. Y. and Singhose, E. W., "Unity-magnitude input shapers and their relation to time-optimal control," in *IFAC World Congress*, (San Francisco, CA), pp. 385–390, 1996.

[63] Pao, L. Y. and Singhose, W. E., "Verifying robust time-optimal commands for multi-mode flexible spacecraft," *J. of Guidance, Control, and Dynamics*, vol. 20, no. 4, pp. 831–833, 1997.

[64] Park, H., Chwa, D., and Hong, K.-S., "A feedback linearization control of container cranes: Varying rope length," *Int. J. of Control, Automation, and Systems*, vol. 5, no. 4, pp. 379–387, 2007.

[65] Park, J. and Chang, P. H., "Learning input shaping technique for non-lti systems," *J. of Dynamic Systems, Measurement, and Control*, vol. 123, pp. 288–293, 2001.

[66] Park, U. H., Lee, J. W., Lim, B. D., and Sung, Y. G., "Design and sensitivity analysis of an input shaping filter in the z-plane," *J. of Sound and Vibration*, vol. 243, no. 1, pp. 157–171, 2001.

[67] Pelez, G. and Singhose, W., "Implementation of input shaping on flexible machines with integer controllers," in *IFAC World Congress*, (Barcelona, Spain), 2002.

[68] Piazzi, A. and Visioli, A., "Optimal dynamic-inversion-based control of an overhead crane," *IEE Proc. Control Theory Applications*, vol. 149, no. 5, pp. 405–411, 2002.

[69] Rappole, B. W., Singer, N. C., and Seering, W. P., "Multiple-mode impulse shaping sequences for reducing residual vibrations," in *23rd Biennial Mechanisms Conf.*, vol. DE-71, (Minneapolis, MN), pp. 11–16, 1994.

[70] Rasmussen, J. and Goodstein, L. P., "Decision support in supervisory control," in *Analysis, Design, and Evaluation of Man-Machine Systems: 2nd IFAC/IFIP/IFORS/IEA Conf.*, (Varese, Italy), 1985.

[71] Rhim, S., *Limitations and Improvements of Time-delay Command Shaping Filters for Flexible Machine Control.* PhD thesis, Georgia Institute of Technology, 2000.

[72] Robertson, M. J. and Singhose, W., "Specified-deflection command shapers for position input systems," *J. of Mechanical Design*, In Press.

[73] Seron, M. M., Goodwin, G. C., and Graebe, S. F., "Control system design issues for unstable linear systems with saturated inputs," *IEE Proc. Control Theory Applications*, vol. 142, no. 4, 1995.

[74] Seth, N., Rattan, K., and Brandstetter, R., "Vibration control of a coordinate measuring machine," in *IEEE Conf. on Control Applications*, (Dayton, OH), pp. 368–73, 1993.

[75] Sheridan, T. B., *Telerobotics, Automation and Human Supervisory Control.* Cambridge, MA: M.I.T. Press, 1992.

[76] Singer, N., Singhose, W., and Kriikku, E., "An input shaping controller enabling cranes to move without sway," in *ANS 7th Topical Meeting on Robotics and Remote Systems*, (Augusta, GA), 1997.

[77] Singer, N. C. and Seering, W. P., "Preshaping command inputs to reduce system vibration," *J. of Dynamic Systems, Measurement, and Control*, vol. 112, no. March, pp. 76–82, 1990.

[78] Singer, N. C., *Redisual Vibration Reduction in Computer Controlled Machines.* PhD thesis, Massachusetts Institute Of Technology, 1989.

[79] Singh, T. and Vadali, S. R., "Robust time-optimal control: A frequency domain approach," *J. of Guidance, Control, and Dynamics*, vol. 17, no. 2, pp. 346–353, 1994.

[80] Singhose, W. E., Seering, W. P., and Singer, N. C., "Time-optimal negative input shapers," *J. of Dynamic Systems, Measurement, and Control*, vol. 119, no. June, pp. 198–205, 1997.

[81] Singhose, W., Banerjee, A., and Seering, W., "Slewing flexible spacecraft with deflection-limiting input shaping," *J. of Guidance, Control, and Dynamics*, vol. 20, no. 2, pp. 291–298, 1997.

[82] Singhose, W., Biediger, E. O., Chen, Y.-H., and Mills, B., "Reference command shaping using specified-negative-amplitude input shapers for vibration reduction," *J. of Dynamic Systems, Measurement, and Control*, vol. 126, no. March, pp. 210–214, 2004.

[83] Singhose, W., Bohlke, K., and Seering, W., "Fuel-efficient shaped command profiles for flexible spacecraft," in *AIAA Guidance, Navigation, and Control Conf.*, (Baltimore, MD), 1995.

[84] Singhose, W., Derezinski, S., and Singer, N., "Extra-insensitive input shapers for controlling flexible spacecraft," *J. of Guidance, Control, and Dynamics*, vol. 19, no. 2, pp. 385–91, 1996.

[85] Singhose, W., Kim, D., and Kenison, M., "Input shaping control of double-pendulum bridge crane oscillations," *J. of Dynamic Systems, Measurement, and Control*, vol. 130, no. 5, 2008.

[86] Singhose, W., Porter, L., Kenison, M., and Kriikku, E., "Effects of hoisting on the input shaping control of gantry cranes," *Control Engineering Practice*, vol. 8, no. 10, pp. 1159–1165, 2000.

[87] Singhose, W., Seering, W., and Singer, N., "Residual vibration reduction using vector diagrams to generate shaped inputs," *J. of Mechanical Design*, vol. 116, no. June, pp. 654–659, 1994.

[88] Singhose, W., Singer, N., and Seering, W., "Improving repeatability of coordinate measuring machines with shaped command signals," *Precision Engineering*, vol. 18, no. April, pp. 138–146, 1996.

[89] Singhose, W. and Sorensen, K., "Manipulation of flexible space structures: Effects of command shaping and human interfaces," in *ICRA Workshop on Space Robotics*, (Rome, Italy), 2007.

[90] Singhose, W. E., Crain, E. A., and Seering, W. P., "Convolved and simultaneous two-mode input shapers," *IEE Control Theory and Applications*, no. Nov., pp. 515–520, 1997.

[91] Singhose, W. E., Porter, L. J., Tuttle, T. D., and Singer, N. C., "Vibration reduction using multi-hump input shapers," *J. of Dynamic Systems, Measurement, and Control*, vol. 119, no. June, pp. 320–326, 1997.

[92] Singhose, W. E., Seering, W. P., and Singer, N. C., "Shaping inputs to reduce vibration: A vector diagram approach," in *IEEE Int. Conf. on Robotics and Automation*, vol. 2, (Cincinnati, OH), pp. 922–927, IEEE, 1990.

[93] Singhose, W. E., Seering, W. P., and Singer, N. C., "Input shaping for vibration reduction with specified insensitivity to modeling errors," in *Japan-USA Symposium on Flexible Automation*, vol. 1, (Boston, MA), pp. 307–313, 1996.

[94] Slaughter, J. B., "Quantization errors in digital control systems," *IEEE Trans. on Automatic Control*, vol. AC-9, pp. 70–74, 1964.

[95] Smith, J. Y., Kozak, K., and Singhose, W. E., "Input shaping for a simple nonlinear system," in *American Control Conf.*, (Anchorage, AK), pp. 821–826, 2002.

[96] Smith, O. J. M., "Posicast control of damped oscillatory systems," *Proceedings of the IRE*, vol. 45, no. September, pp. 1249–1255, 1957.

[97] Sorensen, K., Fisch, H., Dickerson, S., Singhose, W., and Glauser, U., "A multi-operational-mode anti-sway and positioning control for an industrial bridge crane," in *IFAC World Congress*, (Seoul, Korea), 2008.

[98] Sorensen, K. and Singhose, W., "Oscillatory effects of common hard nonlinearities on systems using two-impulse zv input shaping," in *American Control Conf.*, (New York City, NY), pp. 5539–5544, 2007.

[99] Sorensen, K., Singhose, W., and Dickerson, S., "A controller enabling precise positioning and sway reduction in cranes with on-off actuation," in *IFAC World Congress*, vol. 16, (Prague), 2005.

[100] Sorensen, K. L., Andersch, P., and Singhose, W. E., "Analysis and mitigation of rate limiting effects on systems using common input shapers," *Automatica*, Submitted Feb. 2008.

[101] Sorensen, K. L., Cross, P. W., Singhose, W. E., and Prakash, S., "Analysis and mitigation of dead-zone effects on systems using two-impulse zv input shaping," in *ASME IDETC/CIE 6th Int. Conf. on Multibody Systems, Nonlinear Dynamics, and Control*, (Las Vegas, USA), 2007.

[102] Sorensen, K. L., Cross, P. W., Singhose, W. E., and Prakash, S., "R-value analysis and mitigation of dead-zone on systems using two-impulse zv input shaping," *ASME J. of Computational and Nonlinear Dynamics*, Submitted April, 2008.

[103] Sorensen, K. L., Daftari, A., Singhose, W. E., and Hekman, K., "Negative input shaping: Eliminating overcurrenting and maximizing the command space," *J. of Dynamic Systems, Measurement, and Control*, In Press.

[104] Sorensen, K. L., Danielson, J., Dickerson, S., Singhose, W. E., Fisch, H., and Glauser, U., "An industrial bridge crane manipulation system with multiple human interfaces," *Int. J. of Control, Automation, and Systems*, Submitted March, 2008.

[105] Sorensen, K. L., Danielson, J., and Singhose, W. E., "Anti-sway and positioning control for an industrial bridge crane with multi-mode dynamics," in *Int. Symp. on Flexible Automation*, (Atlanta, GA, USA), 2008.

[106] Sorensen, K. L., Hekman, K., and Singhose, W. E., "Finite-state input shaping," *IEEE Trans. on Control Systems Technology*, Submitted March, 2008.

[107] Sorensen, K. L. and Singhose, W. E., "Command-induced vibration analysis using input shaping principles," *Automatica*, In Press.

[108] Sorensen, K. L., Singhose, W. E., and Dickerson, S., "A controller enabling precise positioning and sway reduction in bridge and gantry cranes," *Control Engineering Practice*, vol. 15, no. 7, pp. 825–837, 2007.

[109] Sorensen, K. L., Spiers, J. B., Hall, T., Dickerson, S., and Singhose, W. E., "Implementation and performance of a visual touch-screen interface for crane manipulation," *Int. J. of Control, Automation, and Systems*, Submitted March, 2008.

[110] Sorensen, K. L., Spiers, J. B., and Singhose, W. E., "Operational effects of crane interface devices," in *IEEE Conf. on Industrial Electronics and Applications*, (Harbin, China), pp. 1073–1078, 2007.

[111] Sorensen, K. L., *A Combined Feedback and Command Shaping Controller for Improving Positioning and Reducing Cable Sway in Cranes.* Masters thesis, Georgia Institute of Technology, 2005.

[112] Staehlin, U. and Singh, T., "Design of closed-loop input shaping controllers," in *American Control Conf.*, (Denver, CO), 2003.

[113] Stahre, J., "Evaluating human/machine interaction problems in advanced manufacturing," *Computer Integrated Manufacturing Systems*, vol. 8, no. 2, pp. 143–150, 1995.

[114] Starr, G. P., "Swing-free transport of suspended objects with a path-controlled robot manipulator," *J. of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 97–100, 1985.

[115] Stergiopoulos, J. and Tzes, A., "Adaptive input shaping for nonlinear systems: A case study," in *Mediterranean Conf. on Control and Automation*, (Limassol, Cyprus), 2005.

[116] Sze, T. W. and Calvert, J. F., "Short-time memory devices in closed-loop system - steady-state response," in *AIEE Fall General Meeting*, (Chicago, Ill), pp. 340–344, 1995.

[117] Tallman, G. H. and Smith, O. J. M., "Analog study of dead-beat posicast control.," *IRE Trans. on Automatic Control*, pp. 14–21, 1958.

[118] Tao, G. and Kokotovic, P. V., "Continuous-time adaptive control of systems with unknown backlash," *IEEE Trans. on Automatic Control*, vol. 40, no. 6, 1995.

[119] Tian, M. and Tao, G., "Adaptive output dead-zone compensation," in *IEEE Conf. on Decision and Control*, (San Diego, CA), pp. 1157 – 1161, 1997.

[120] Tuttle, T. D. and Seering, W. P., "Vibration reduction in flexible space structures using input shaping on mace: Mission results," in *IFAC World Congress*, (San Francisco, CA), 1996.

[121] Tzeng, C.-Y. and Lin, K.-F., "Adaptive ship steering autopilot design with saturating and slew rate limiting actuator," *Int. J. of Adaptive Control and Signal Processing*, no. 14, pp. 411–426, 2000.

[122] Tzes, A., "Robust control design issues for input-shaped discrete systems," in *American Control Conf.*, (Denver, CO), pp. 4785–4787, 2003.

[123] Tzes, A. and Yurkovich, S., "An adaptive input shaping control scheme for vibration suppression in slewing flexible structures," *IEEE Trans. on Control Systems Technology*, vol. 1, no. June, pp. 114–121, 1993.

[124] Vaughan, J., Kim, D., and Singhose, W., "Control of tower cranes with double-pendulum payload dynamics," *Submitted to IEEE Trans. on Control Systems Technology*, 2008.

[125] Zuo, K., Drapeau, V., and Wang, D., "Closed loop shaped-input strategies for flexible robots," *Int. J. of Robotics Research*, vol. 14, no. 5, pp. 510–529, 1995.

[126] Zuo, K. and Wang, D., "Closed loop shaped-input control of a class of manipulators with a single flexible link," in *IEEE Int. Conf. on Robotics and Automation*, vol. 1, (Nice, France), pp. 782–787, IEEE, 1992.